

10/525033

CT/JP03/10499

10 Rec'd PCT

17 FEB 2005

20.08.03

日本国特許庁
JAPAN PATENT OFFICE

REC'D 19 OCT 2003

WIPO PCT

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日 2002年 8月20日
Date of Application:

出願番号 特願2002-238710
Application Number:
[ST. 10/C]: [JP 2002-238710]

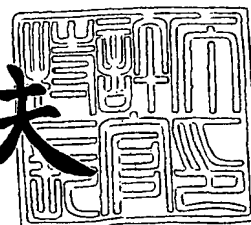
出願人 江良 一成
Applicant(s):

PRIORITY DOCUMENT
SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH
RULE 17.1(a) OR (b)

2003年 9月25日

特許庁長官
Commissioner,
Japan Patent Office

今井康夫



Best Available Copy

出証番号 出証特2003-307839

【書類名】 特許願

【整理番号】 MQ02002

【提出日】 平成14年 8月20日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 15/62

【発明の名称】 画像表示装置

【請求項の数】 3

【発明者】

 【住所又は居所】 千葉県柏市松葉町 4-1-11-201

 【氏名】 江良 一成

【特許出願人】

 【識別番号】 596030737

 【住所又は居所】 千葉県柏市松葉町 4-1-11-201

 【氏名又は名称】 江良 一成

【代理人】

 【識別番号】 100083231

 【住所又は居所】 東京都港区新橋 2丁目 10番 5号 末吉ビル 5階 ミネ
ルバ国際特許事務所

 【弁理士】

 【氏名又は名称】 紋田 誠

【手数料の表示】

 【予納台帳番号】 016241

 【納付金額】 21,000円

【提出物件の目録】

 【物件名】 明細書 1

 【物件名】 図面 1

 【物件名】 要約書 1

 【包括委任状番号】 9810475

【ブルーフの要否】 要

【書類名】 明細書

【発明の名称】 画像表示装置

【特許請求の範囲】

【請求項 1】 立体視画像表示機能を備えた画像表示装置において、
非立体視画像表示モードと立体視表示モードの切換スイッチと、
立体視画像データを作成する立体視画像データ作成手段を備え、
上記切換スイッチにより非立体視画像表示モードが選択されたときには、表示画像データを非立体視画像表示モードで表示する一方、上記切換スイッチにより立体視画像表示モードが選択されたときには、表示画像データを上記立体視画像データ作成手段により立体視画像データに変換し、その立体視画像データを立体視表示モードで表示するようにしたことを特徴とする画像表示装置。

【請求項 2】 立体視画像表示機能を備えた画像表示装置において、
非立体視画像表示モードと立体視表示モードの切換スイッチと、
立体視画像データを作成する立体視画像データ作成手段を備え、
上記切換スイッチにより非立体視画像表示モードが選択されたときには、表示画像データを非立体視画像表示モードで表示する一方、
上記切換スイッチにより立体視画像表示モードが選択されたときには、表示画像データを上記立体視画像データ作成手段により左目用画像データと右目用画像データに変換して、左目用画像データと右目用画像データをそれぞれ左目用画像データ領域と右目用画像データ領域に保存し、
上記左目用画像データ領域に保存した左目用画像データと、右目用画像データ領域に保存した右目用画像データを、交互に表示装置へ表示させることを特徴とした画像表示装置。

【請求項 3】 前記立体視画像データ作成手段は、入力した表示画像データの 1 画面分の奥行き値の平均値と、上記二次元画像データの複数サンプリング領域の奥行き値との関係に基づいて、上記二次元画像データであらわされた 1 画面の明るさの傾向が、画面の下から上方向へ増大しているか、あるいは、中央部から周辺部へ向かって増大しているか、もしくは、画面の上から下方向へ増大しているか、あるいは、中央部から周辺部へ向かって減少しているかを判断し、前者

の場合には、上記奥行き値の変化傾向と距離の変化傾向が同一であると判断する一方、後者の場合には、上記奥行き値の変化傾向に対して距離の変化傾向が反転していると判断する奥行き値反転判断手段と、

上記二次元画像データのそれぞれの画素について、1画面分の奥行き値の平均値と、奥行き値の最大値と、奥行き値の最小値に基づいて、それぞれの画素について、奥行き値を、その分布が拡張する態様に修正した修正奥行き値を算出する第1の奥行き値修正手段と、

上記二次元画像データの1画面に含まれる1つ以上の独立物体を抽出し、その独立物体の周縁部を結ぶ等高線をおのおのの奥行き値について形成するとともに、同一奥行き値を有する等高面を決定し、その等高面の奥行き値を、あらかじめ設定した距離条件に従って算出する参照奥行き値算出手段と、

おのおのの画素について、当該画素が含まれる上記等高面での参照奥行き値の変化態様が、上記修正奥行き値の変化態様と異なっている場合には、修正奥行き値の順序が保存される態様にその修正奥行き値を修正し、第2の修正奥行き値を算出する第2の奥行き値修正手段と、

おのおのの画素について、上記第2の修正奥行き値に基づき所定の丸め係数を用いて丸め奥行き値を算出し、その丸め奥行き値をあらかじめ設定された上限値と下限値で定められる範囲の値に制限して第2の丸め奥行き値を算出する適用奥行き値算出手段と、

処理対象となる画素について、上記奥行き値反転判断手段で上記奥行き値の変化傾向と距離の変化傾向が同一であると判断されている場合には、当該画素について算出された上記第2の丸め奥行き値に所定の奥行き深度を加えた値を視差値として算出する一方、上記奥行き値反転判断手段で上記奥行き値の変化傾向と距離の変化傾向が反転していると判断されている場合には、当該画素について算出された上記第2の丸め奥行き値を、奥行き値の取りうる最大値より減じた値に、上記奥行き深度を加えた値を、当該画素の視差値として算出する視差値算出手段と、

処理対象となる画素について、上記視差値算出手段で算出された視差値、および、上記二次元画像データにおける原座標に基づいて、当該画素の画像データを、右目用画像データまたは左目用画像データのそれぞれの記憶領域の対応する座

標領域に保存する画像データ作成手段を備えたことを特徴とする請求項1または請求項2記載の画像表示装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、立体視画像表示機能を備えた画像表示装置に関する。

【0002】

【従来の技術】

従来より、写真または映画を立体的な画像または映像として認識させる立体視を利用して、写真または映画を鑑賞する鑑賞方法が実用されている。

【0003】

ここで、立体視とは、『視覚、特に両眼を用いた形態視において、網膜に投影された平面像に基づく感覚情報が、大脳皮質の視覚野における統合作用を受け、三次元空間内の立体として認知されることをいう。実体視ともいわれる。立体視は、実際の立体を見る場合だけでなく、左右の目にそれぞれ一定の規則に従って異なる平面図形を見せた場合にも生ずる。立体写真や、立体図形を実体鏡で見た場合がこれにあたる。このことから、立体視には両眼の間の視差が重要であることがわかる。しかし単眼視でも、水晶体の曲率を調節している毛様体筋の収縮の程度や、目の位置を変えたときの視差などにより、ある程度の遠近や奥行き判断が可能である。霊長類では、視神経の半分が交差し、左右両眼の右半分の網膜からの神経繊維は右側皮質後頭葉の視覚野に達する。同様に、網膜の左半分が左側皮質後頭葉に連絡する。視覚野のニューロンは網膜の一定の場所に対応した入力を受けている。しかも視覚野のニューロンの大多数には、いろいろな程度に両眼からの入力があり、これが立体視に関係するといわれている。両眼視では、左右の目で見ている1つの物体は、1つに見える。眼球は、左右の網膜の像がそれぞれ対応する点にうまく合致して1つの像として見えるように反射的に運動する。これが遠近感を作る1つの原因となっている。空間知覚は、右脳に発達するので、視野の左半分の方が物体の空間的位置関係がよく認知されると言われている。』（「日本大百科全書(小学館)」より抄出)

【0004】

したがって、立体視用の2枚1組の写真を写真を撮影するためには、同等な2つの撮影用レンズを両眼の間隔（例えば、6.5mm程度）離し、それぞれの光軸が平行になるように設置し、それぞれの撮影用レンズのシャッターを同時に作用して、2コマのフィルムに同時に映像を撮影する。そして、その2つのフィルムを現像し、印画紙に映像を焼き付けて、立体視用の写真画像を形成する。

【0005】

この2枚の立体視用写真画像を鑑賞するには、例えば、専用のビューワーを用いて2枚の立体視用写真画像を左右の目でそれぞれ鑑賞する。それにより、撮影対象が立体的に認識される。

【0006】

また、立体視用の映画を撮影するときには、所定間隔を離して設置した2台のカメラを用いて映像を撮影し、2本1組の映画フィルムを得る。そして、それぞれの映画フィルムを、偏光方向が互いに直行する2枚の偏光フィルタを映写レンズの直後に備えた2つの映写機で、同一のスクリーンに同時に映写する。

【0007】

そして、鑑賞する人は、偏光方向が互いに直行する2枚の偏光フィルタがそれぞれ両眼に対応して配設された鑑賞用眼鏡を用いてスクリーンに映写された映像を鑑賞すると、映像を立体的に認識することができる。このとき、鑑賞用眼鏡に配設される左目用フィルタおよび右目用フィルタは、左目用映像を映写する映写機の映写レンズおよび右目用映像を映写する映写機の映写レンズに設けられたと同種の偏光フィルタが配設される。

【0008】

しかしながら、このような従来の立体視用写真または立体視用映画を撮影する際には、その撮影時に、撮影する場所において、立体視用写真または立体視用映画を撮影する必要がある、時間的場所的な制約があった。

【0009】

そこで、静止画像から立体視画像を作成するものとして、例えば、特開2002-123842号「立体視画像生成装置および情報記憶媒体」が提案されてい

る。

【0010】

この従来技術を利用することで、非立体視画像データをリアルタイムに立体視画像データに変換することができ、上述した時間的場所的な制約に縛られなく、自由に立体視画像を表示できるようになった。

【0011】

【発明が解決しようとする課題】

ところで、従来、立体視画像表示装置では、常時、立体視画像を表示しているため、非立体視画像を表示する際には、面倒な操作が必要であった。

【0012】

本発明は、かかる実情に鑑みてなされたものであり、立体視画像と非立体視画像を適宜に表示することのできる画像表示装置を提供することを目的としている。

【0013】

【課題を解決するための手段】

本発明は、立体視画像表示機能を備えた画像表示装置において、非立体視画像表示モードと立体視表示モードの切換スイッチと、立体視画像データを作成する立体視画像データ作成手段を備え、上記切換スイッチにより非立体視画像表示モードが選択されたときには、表示画像データを非立体視画像表示モードで表示する一方、上記切換スイッチにより立体視画像表示モードが選択されたときには、表示画像データを上記立体視画像データ作成手段により立体視画像データに変換し、その立体視画像データを立体視表示モードで表示するようにしたものである。

【0014】

また、立体視画像表示機能を備えた画像表示装置において、非立体視画像表示モードと立体視表示モードの切換スイッチと、立体視画像データを作成する立体視画像データ作成手段を備え、上記切換スイッチにより非立体視画像表示モードが選択されたときには、表示画像データを非立体視画像表示モードで表示する一方、上記切換スイッチにより立体視画像表示モードが選択されたときには、表示

画像データを上記立体視画像データ作成手段により左目用画像データと右目用画像データに変換して、左目用画像データと右目用画像データをそれぞれ左目用画像データ領域と右目用画像データ領域に保存し、上記左目用画像データ領域に保存した左目用画像データと、右目用画像データ領域に保存した右目用画像データを、交互に表示装置へ表示させるようにしたものである。

【0015】

また、前記立体視画像データ作成手段は、入力した表示画像データの1画面分の奥行き値の平均値と、上記二次元画像データの複数サンプリング領域の奥行き値との関係に基づいて、上記二次元画像データであらわされた1画面の明るさの傾向が、画面の下から上方向へ増大しているか、あるいは、中央部から周辺部へ向かって増大しているか、もしくは、画面の上から下方向へ増大しているか、あるいは、中央部から周辺部へ向かって減少しているかを判断し、前者の場合には、上記奥行き値の変化傾向と距離の変化傾向が同一であると判断する一方、後者の場合には、上記奥行き値の変化傾向に対して距離の変化傾向が反転していると判断する奥行き値反転判断手段と、上記二次元画像データのそれぞれの画素について、1画面分の奥行き値の平均値と、奥行き値の最大値と、奥行き値の最小値に基づいて、それぞれの画素について、奥行き値を、その分布が拡張する態様に修正した修正奥行き値を算出する第1の奥行き値修正手段と、上記二次元画像データの1画面に含まれる1つ以上の独立物体を抽出し、その独立物体の周縁部を結ぶ等高線をおのおのの奥行き値について形成するとともに、同一奥行き値を有する等高面を決定し、その等高面の奥行き値を、あらかじめ設定した距離条件に従って算出する参照奥行き値算出手段と、おのおのの画素について、当該画素が含まれる上記等高面での参照奥行き値の変化態様が、上記修正奥行き値の変化態様と異なっている場合には、修正奥行き値の順序が保存される態様にその修正奥行き値を修正し、第2の修正奥行き値を算出する第2の奥行き値修正手段と、おのおのの画素について、上記第2の修正奥行き値に基づき所定の丸め係数を用いて丸め奥行き値を算出し、その丸め奥行き値をあらかじめ設定された上限値と下限値で定められる範囲の値に制限して第2の丸め奥行き値を算出する適用奥行き値算出手段と、処理対象となる画素について、上記奥行き値反転判断手段で上記

奥行き値の変化傾向と距離の変化傾向が同一であると判断されている場合には、当該画素について算出された上記第2の丸め奥行き値に所定の奥行深度を加えた値を視差値として算出する一方、上記奥行き値反転判断手段で上記奥行き値の変化傾向と距離の変化傾向が反転していると判断されている場合には、当該画素について算出された上記第2の丸め奥行き値を、奥行き値の取りうる最大値より減じた値に、上記奥行深度を加えた値を、当該画素の視差値として算出する視差値算出手段と、処理対象となる画素について、上記視差値算出手段で算出された視差値、および、上記二次元画像データにおける原座標に基づいて、当該画素の画像データを、右目用画像データまたは左目用画像データのそれぞれの記憶領域の対応する座標領域に保存する画像データ作成手段を備えたものである。

【0016】

【発明の実施の形態】

以下、添付図面を参照しながら、本発明の実施の形態を詳細に説明する。

【0017】

図1は、本発明の一実施例にかかる移動体通信端末装置を示している。

【0018】

同図において、CPU（中央処理装置）1は、この移動体通信端末装置の動作制御、立体視用画像データ作成処理などの各種処理を実行するためのものであり、ROM（リード・オンリ・メモリ）2は、起動時などに実行されるプログラムや参照データなどが記憶されるものであり、RAM（ランダム・アクセス・メモリ）3は、CPU1のワークエリアなどを実現するためのものであり、キャラクタージェネレータ4は、図形文字の表示データ（フォントデータ）を発生するものであり、時計回路5は、現在時刻情報を出力するためのものである。

【0019】

無線通信ユニット6は、この移動体通信端末装置を移動体通信網へ接続するためのものであり、通話ユニット7は、ユーザが通話するためのもの（スピーカユニットおよびマイクユニットを含む）である。

【0020】

液晶表示器8は、非立体視モードの画像表示機能と立体視モードの画像表示機

能を備えたものであり、裸眼立体視が可能な表示画面を備えている。また、表示制御部 9 は、液晶表示器 8 の表示内容を制御するためのものである。そして、液晶表示器 8 および表示制御部 9 により、立体視表示装置 1 0 が構成される。

【0 0 2 1】

入力操作部 1 1 は、テンキー等、移動体通信端末装置に必要な種々の入力操作を行うためのものであり、入力制御部 1 2 は、入力操作部 1 1 の入力データを取り込むためのものである。

【0 0 2 2】

これらの CPU 1、ROM 2、RAM 3、キャラクタジェネレータ 4、時計回路 5、無線通信ユニット 7、通話ユニット 7、表示制御部 8、および、入力制御部 1 2 は、内部バス 1 3 に接続されており、これらの各種要素間のデータのやりとりは、主としてこの内部バス 1 3 を介して行われている。

【0 0 2 3】

図 2 は、移動体通信端末装置の筐体の概略構成を示している。

【0 0 2 4】

同図において、筐体の下側には、電話番号等を入力するためのテンキー T K が設けられており、その上側には、ファンクションキー F K および表示モードを非立体視モード（2 D モード）と立体視モード（3 D モード）に切換えるための切換スイッチ D K が設けられている。

【0 0 2 5】

また、上部には、液晶表示器 8 の表示画面が配設されており、筐体の上側には、アンテナ A T が設けられる。また、通話ユニット 7 のスピーカユニットのための孔部 7 a と、マイクユニットのための孔部 7 b がそれぞれ適宜な場所に設けられている。

【0 0 2 6】

図 3 は、この移動体通信端末装置で、静止画を表示する場合の処理の一例を示している。

【0 0 2 7】

まず、立体視表示装置 1 0 を非立体視モード（2 D モード）に設定し（処理 S

101)、非立体視モードの静止画データを立体視表示装置10へ出力する(処理S102)。

【0028】

これにより、初期状態では、移動体通信端末装置のユーザは、非立体視モードの静止画を見ることができる。

【0029】

その状態で、ユーザが切換スイッチDKを切換操作するか、あるいは、画像表示の終了が指定されるまで待つ(判断S103, S104のNOループ)。画像表示の終了が指定された場合で、判断S104の結果がYESになるときは、画像表示を終了し(処理S105)、この表示動作を終了する。

【0030】

また、切換スイッチDKが切換操作された場合で、判断S103の結果がYESになるときは、立体視表示装置10に立体視表示モードをセットし(処理S106)、所定の立体視表示データ作成処理を行い(処理S107)、それによって得た立体視表示データを、立体視表示装置10へ出力する(処理S108)。

【0031】

これにより、ユーザは、立体視画像を見ることができる。

【0032】

また、この状態で、ユーザが切換スイッチDKを切換操作するか、あるいは、画像表示の終了が指定されるまで待つ(判断S109, S110のNOループ)。画像表示の終了が指定された場合で、判断S110の結果がYESになるときは、画像表示を終了し(処理S111)、この表示動作を終了する。

【0033】

また、切換スイッチDKが切換操作された場合で、判断S109の結果がYESになるときは、処理S101へ戻り、立体視表示装置10を非立体視モード(2Dモード)に設定して、それ以降の処理を実行する。

【0034】

図4は、この移動体通信端末装置で、動画を受信して表示する場合の処理の一

例を示している。

【0035】

まず、立体視表示装置10を非立体視モード（2Dモード）に設定し（処理S201）、動画データを受信し（処理S202）、受信した動画データの1フレーム分の画像データを立体視表示装置10へ出力する（処理S203）。

【0036】

次いで、受信した動画データの次のフレームがあるかどうかを調べ（判断S204）、判断S204の結果がNOになるときには、表示終了して（処理S205）、この処理を終了する。

【0037】

また、判断S204の結果がYESになるときには、ユーザが切換スイッチDKを切換操作するか、あるいは、画像表示の終了が指定されるまで待つ（判断S206, S207）。

【0038】

画像表示の終了が指定された場合で、判断S207の結果がYESになるときには、処理S205へ移行し、画像表示を終了して、この表示動作を終了する。また、判断S207の結果がNOになるときには、処理S202へ戻り、次のフレームの表示動作を行う。

【0039】

これにより、初期状態では、移動体通信端末装置のユーザは、非立体視モードの動画を見ることができる。

【0040】

また、切換スイッチDKが切換操作された場合で、判断S206の結果がYESになるときには、立体視表示装置10に立体視表示モードをセットし（処理S208）、動画データを受信し（処理S209）、受信した動画データの1フレーム分の画像データを画像バッファへ取り込み（処理S210）、その1フレーム分の画像データについて所定の立体視表示データ作成処理を行い（処理S211）、それによって得た立体視表示データを立体視表示装置10へ出力する（処理S212）。

【0041】

これにより、ユーザは、立体視表示モードの動画を見ることができる。

【0042】

また、この状態で、受信した動画データの次のフレームがあるかどうかを調べ（判断S213）、判断S213の結果がNOになるときには、処理S205へ移行し、表示終了して、この処理を終了する。

【0043】

また、判断S213の結果がYESになるときには、ユーザが切換スイッチDKを切換操作するか、あるいは、画像表示の終了が指定されるのを調べる（判断S214、S215）。判断S215の結果がNOになるときには、処理S209へ戻り、動画データの立体視表示モードの動作を継続する。

【0044】

また、表示終了が指令された場合で、判断S215の結果がYESになるときには、処理S205へ移行し、表示終了して、この処理を終了する。

【0045】

また、切換スイッチDKが切換操作された場合で、判断S214の結果がYESになるときには、処理S201へ戻り、立体視表示装置10を非立体視モード（2Dモード）に設定して、それ以降の処理を実行する。

【0046】

このようにして、本実施例では、ユーザが切換スイッチDKを切換操作するたびに、液晶表示器8に表示される画像（静止画または動画）が、非立体視モードの静止画から立体視表示モードの静止画像に、また、立体視表示モードの静止画から非立体視表示モードの静止画像に切り替わるので、非常に便利である。

【0047】

次に、立体視表示データ作成処理（処理S107、S211）の一例について説明する。

【0048】

本実施例では、立体視を効果的に実現できるための種々の工夫をこらしており、その背景について説明する。

【0049】

立体視は、両眼を備える人間が、左右それぞれの目で見える映像が別々に脳に到達し、処理されることで認識される感覚の1つである。

【0050】

しかしながら、我々が普通の映像（通常の二次元映像等の非立体視映像）を見ても遠近感や立体感を感知できており、映像を楽しむことができる。

【0051】

これらのことから、立体視には、視覚からの映像の処理だけでなく、映像を認識する際の学習効果からも大きい影響があるものと考えられる。例えば、だまし絵の多くは、立体視に関する学習効果を逆手に取り、見ている人が立体的な効果を体得できるような表現をしたものである。

【0052】

本発明では、単に視覚から受ける視差のみで立体感を構成するのではなく、1物体上のZ値（奥行き値）の変化を滑らかにし、物体と背景との境界の輪郭部のZ値の変化にメリハリを付けることで、より自然な立体感を感じるように工夫している。

【0053】

一般に、普通の視力をもつ人間が、視覚により立体感を識別できるのは、約140（メートル）程度である。我々は、風景を見るとき、立体視できる映像（近景）と、立体視できない映像（遠景）の混在した映像で、遠近感を感知していることとなる。

【0054】

ところが、かかる事情があっても、通常の状態では、映像に違和感がなく、風景を認識している。

【0055】

すなわち、140（メートル）を超えた遠景の部分については、われわれは、立体に対する学習効果に基づいて、立体を認識していると考えられる。

【0056】

このようなことを背景に、本発明は、人間が効果的に立体視を感知できるよう

な立体視画像データを作成する方法を工夫している。

【0057】

すなわち、換言すれば、本発明は、単なる数値演算結果により視差を求めるだけではなく、人間が強く立体感を感じる場合にはどのような条件が必要であるかの研究に基づいてなされたものである。

【0058】

かかる研究によれば、物体と物体とが重なり合う部分に視差の違いが顕著に存する場合に、人間はより強く立体感を知覚できるとの結論に達した。

【0059】

しかしながら、物体と物体が重なり合う部分の多くには、強い光源からの光が当たり、ハイライト部分が形成されて、強い視差が形成され、その結果、物体と物体の重なり合う部分に画像の乱れが生じやすく、深い視差を与えるには限界があった。

【0060】

このような限界を乗り越え、より深い視差を求めるためには、画像の乱れの原因となる局所的な強いZ値や極端に低いZ値を、違和感や立体感の低下が生じないように補正できれば良いこととなる。

【0061】

そのため、本発明は、次のような種々の補正手段を創出した。

【0062】

①暗部におけるZ値の拡張、②輪郭抽出、③丸め係数、④画像の特質の判定、⑤上限基準・下限基準による高Z値・低Z値の補正、⑥メッシュZ値によるZ値順序の補正。

【0063】

ここで、以下の記述に出てくる用語の説明を行う。

【0064】

まず、「奥行深度」とは、画像表示のインタフェース（画面）から立体世界が、どの位置に表示されるかの深さを示す。例えば、左目画像について視差分を左に移動し、右目画像について視差分を右に移動して表示された立体視画像が、表

示画面から内部に引いて見えている状態であるとすれば、視差分の移動方向を逆転することで、立体視画像は、表示画面から飛び出して見える。

【0065】

また、「オブジェクト深度」とは、立体視世界の最低奥行から最高奥行までの深さを示す。このパラメータの値を大きくすると、立体感が増加することとなる。

【0066】

また、「奥行き値（Z値）」とは、画像に含まれる物体に割り当てる（画面に対する）奥行方向（Z方向）の距離をあらわしている。

【0067】

図5は、立体視表示データ作成処理（処理S107, S211）の一例を示している。

【0068】

まず、処理対象となっている1画面分のカラー画像データ（＝二次元カラー画像データ；処理S107の場合は画像1枚分の静止画データ、処理S211の場合は動画データの1フレーム分の画像データ）を読み込み（処理101）、そのカラー画像データがRGBフォーマットのものであるかどうかを調べる（判断102）。

【0069】

判断102の結果がNOになるとときには、処理101で読み込んだ1画面分のカラー画像データを、RGBフォーマットのカラー画像データへ変換する（処理103）。

【0070】

ここで、カラー画像データの1画素あたりの色彩深度としては、16ビット、24ビット、または、32ビットのいずれかのものを適用することができる。

【0071】

そして、その1フレーム分のカラー画像データについて、まず、それぞれの画素について、画像データ値に基づいてグレースケール値を算出し、その算出したグレースケール値をZ値（奥行き値）とするZ値算出処理（処理104（＝グ

ースケール値算出ステップ)) を行う。

【0072】

次に、カラー画像データのそれぞれの画素について算出したグレースケール値の平均値と、カラー画像データの複数サンプリング領域のグレースケール値との関係に基づいて、その画面の明るさの傾向が、画面の下から上方向へ増大しているか、あるいは、中央部から周辺部へ向かって増大しているか、もしくは、画面の上から下方向へ増大しているか、あるいは、中央部から周辺部へ向かって減少しているかを判断し、前者の場合には、グレースケール値の変化傾向と奥行き値の変化傾向が同一であると判断する一方、後者の場合には、グレースケール値の変化傾向に対して奥行き値の変化傾向が反転していると判断する Z 値反転判断処理 (処理 105 (=奥行き値反転判断ステップ)) を行う (上述した④に関連する処理を含む)。

【0073】

次いで、カラー画像データのそれぞれの画素について、当該注目画素 (処理対象となっている画素; 以下同じ) を含む所定サイズの二次元マトリクスに含まれるグレースケール値の平均値を、当該画素の奥行き値として割り当てる輪郭抽出基礎データ処理 (処理 106 (=奥行き値算出ステップ)) を行う (上述した②に関する処理を含む)。

【0074】

次に、カラー画像データのそれぞれの画素について処理 106 で算出した奥行き値の画面全体の平均値と、最大値と、最小値に基づいて、それぞれの画素について、奥行き値を修正する Z 値拡張処理 (処理 107 (=第 1 の奥行き値修正ステップ)) を行う (上述した①に関する処理)。

【0075】

次に、カラー画像データに含まれる 1 つ以上の独立物体を抽出し、その抽出した独立物体に含まれる等高線 (後述) について、新たに奥行き値を算出し、その新たに算出した奥行き値に基づいて、奥行き値修正値を算出する参照 Z 値算出処理 (処理 108 (=参照奥行き値算出ステップ)) を行う (上述した③, ⑥に関する処理を含む)。

【0076】

次いで、おのおのの画素について、処理108で算出した奥行き値に、上記参照奥行き値算出ステップで算出した奥行き値修正値を加算して、奥行き値を修正するZ値修正処理（処理109（＝第2の奥行き値修正ステップ））を行う。

【0077】

次に、おのおのの画素について、それまでに算出して得られたデータに基づいて、左目用画像データおよび右目用画像データを作成して出力する立体変換出力処理（処理110（＝適用奥行き値算出ステップ、視差値算出ステップ、および、画像データ作成ステップ））を行う。

【0078】

次に、各処理について具体的に説明する。なお、以下の説明において、式の構文やデータ型等の記述方法は、C言語またはその拡張言語（C++言語等）の記述方法に準拠している。

【0079】

上述したように、本実施例では、色彩深度として16ビット、24ビット、および、32ビットのいずれかのRGBフォーマットのカラー画像データを用いることができる。

【0080】

ここで、色彩深度が16ビットのカラー画像データの1画素のデータは、図6(a)に示すように、Rデータが5ビット、Gデータが6ビット、および、Bデータが5ビットのデータ構成を持つ。

【0081】

また、色彩深度が24ビットのカラー画像データの1画素のデータは、同図(b)に示すように、Rデータが8ビット（＝1バイト）、Gデータが8ビット、および、Bデータが8ビットのデータ構成を持つ。

【0082】

また、色彩深度が32ビットのカラー画像データの1画素のデータは、同図(c)に示すように、24ビットのカラー画像データの先頭に8ビットのダミーデータを付加したデータ構成を持つ。

【0083】

図7, 8, 9は、Z値算出処理（処理104）の一例を示している。

【0084】

このZ値算出処理では、それぞれの画素について、Rデータの30%と、Gデータの59%と、Bデータの11%の総和を取った値を、8ビットのグレースケール値として算出し、そのグレースケール値を当該画素のZ値（*zinfo*）として算出している。なお、Rデータ、Gデータ、および、Bデータの重み付けの値（30%、59%、11%）は、人間の網膜に存在する赤錐体、緑錐体、および、青錐体の数の分布に基づき、立体視画像を作成する際に特に適する値を設定したものである。

【0085】

ここで、色彩深度が16ビットの場合には、グレースケール値の最大値は31であり、色彩深度が24ビットおよび32ビットの場合には、グレースケール値の最大値は255である。

【0086】

ここで、**src*は元画像データが格納された領域へのポインタ、WDは元画像の横幅画素数、HTは元画像の縦幅画素数、**zinfo*は計算されたZ値を格納する領域へのポインタをあらわす。

【0087】

まず、色彩深度が16ビットであるか否かを調べる（判断201）。判断201の結果がYESになるときは、「*Unsigned Short *srcp*」を変数宣言し（処理202）、注目画素のデータ*src*を*unsigned short*へ型変換し（処理203）、変数*srcp*に、データ*src*を代入する（処理204）。

【0088】

次いで、カウンタCCの値を0に初期設定し（処理205）、カウンタCCの値が、「WD*HT」よりも小さいかどうかを調べる（判断206）。判断206の結果がNOになるときは、1画面分の画素についての処理が終了したので、この処理を終了する。

【0089】

また、判断206の結果がYESになるとときには、1画面分の画素についての処理が終了していない場合である。

【0090】

この場合には、変数Rに、ポインタ*srcpに格納されているデータ（16ビットデータ）と16進数「7c00」を論理積した結果を10ビット算術右シフトし、その結果に30を乗じた値を、intに型変換した結果の値を代入する（処理207）。それにより、変数Rには、Rデータ（5ビット）に30を乗じた値が代入される。

【0091】

次に、変数Gに、ポインタ*srcpに格納されているデータ（16ビットデータ）と16進数「03e0」を論理積した結果を5ビット算術右シフトし、その結果に59を乗じた値を、intに型変換した結果の値を代入する（処理208）。それにより、変数Gには、Gデータ（6ビット）に59を乗じた値が代入される。

【0092】

次に、変数Bに、ポインタ*srcpに格納されているデータ（16ビットデータ）と16進数「001f」を論理積した結果に11を乗じた値を、intに型変換した結果の値を代入する（処理209）。それにより、変数Bには、Bデータ（5ビット）に11を乗じた値が代入される。

【0093】

そして、変数Rと変数Gと変数Bの値の総和を100で除算した結果を、ポインタ*zinfoであらわされる領域へ代入する（処理210）。

【0094】

これで、1画素分の処理が終了したので、変数srcp, zinfo、カウンタCCの値をそれぞれ1つ増やして（処理211）、判断206へ戻り、それ以降の処理を行う。

【0095】

また、判断201の結果がNOになるとときには、色彩深度が24ビットである

か否かを調べる（判断 215）。判断 215の結果がYESになるとときには、カウンタCCの値を0に初期設定し（処理 216）、カウンタCCの値が、「WD*HT」よりも小さいかどうかを調べる（判断 217）。判断 217の結果がNOになるとときには、1画面分の画素についての処理が終了したので、この処理を終了する。

【0096】

また、判断 217の結果がYESになるとときには、1画面分の画素についての処理が終了していない場合である。

【0097】

この場合には、変数Bに、ポインタ*srcに格納されているデータに11を乗じた値を代入する（処理 218）。次に、変数Gに、ポインタ*（src+1）に格納されているデータに59を乗じた値を代入する（処理 219）。次いで、変数Rに、ポインタ*（src+2）に格納されているデータに30を乗じた値を代入する（処理 220）。

【0098】

そして、変数Rと変数Gと変数Bの値の総和を100で除算した結果を、ポインタ*zinfoであらわされる領域へ代入する（処理 221）。

【0099】

これで、1画素分の処理が終了したので、変数srcp, zinfo、カウンタCCの値をそれぞれ1つ増やして（処理 222）、判断 217へ戻り、それ以降の処理を行う。

【0100】

また、判断 215の結果がNOになるとときには、色彩深度が32ビットの場合である。この場合には、「Unsigned long *srcp」を変数宣言し（処理 225）、注目画素のデータsrcをunsigned longへ型変換し（処理 226）、変数srcpに、データsrcを代入する（処理 227）。

【0101】

次いで、カウンタCCの値を0に初期設定し（処理 228）、カウンタCCの

値が、「WD*HT」よりも小さいかどうかを調べる（判断229）。判断229の結果がNOになるときには、1画面分の画素についての処理が終了したので、この処理を終了する。

【0102】

また、判断229の結果がYESになるときには、1画面分の画素についての処理が終了していない場合である。

【0103】

この場合には、変数Rに、ポインタ*srcpに格納されているデータ（32ビットデータ）と16進数「00ff0000」を論理積した結果を16ビット算術右シフトし、その結果に30を乗じた値を、intに型変換した結果の値を代入する（処理230）。それにより、変数Rには、Rデータ（8ビット）に30を乗じた値が代入される。

【0104】

次に、変数Gに、ポインタ*srcpに格納されているデータ（32ビットデータ）と16進数「0000ff00」を論理積した結果を5ビット算術右シフトし、その結果に59を乗じた値を、intに型変換した結果の値を代入する（処理231）。それにより、変数Gには、Gデータ（8ビット）に59を乗じた値が代入される。

【0105】

次に、変数Bに、ポインタ*srcpに格納されているデータ（32ビットデータ）と16進数「0000ff」を論理積した結果に11を乗じた値を、intに型変換した結果の値を代入する（処理232）。それにより、変数Bには、Bデータ（8ビット）に11を乗じた値が代入される。

【0106】

そして、変数Rと変数Gと変数Bの値の総和を100で除算した結果を、ポインタ*zinfoであらわされる領域へ代入する（処理233）。

【0107】

これで、1画素分の処理が終了したので、変数srcp, zinfo、カウンタCCの値をそれぞれ1つ増やして（処理234）、判断228へ戻り、それ以

降の処理を行う。

【0108】

次に、Z値反転判定処理（処理105）について説明する。

【0109】

このZ値反転判定処理では、立体視をした際に、Z値が大きいほど、遠くにあるような立体感を得ることができるようにするため、カラー画像の傾向を調べ、Z値を反転修正するか否かを判定している。

【0110】

まず、図10に示すように、カラー画像データの1画面の左上隅、中央上端、右上隅、中央左端、中央、中央右端、左下隅、中央下端、右下隅に、それぞれ縦8ドットで横8ドットのサイズの二次元マトリクス領域を設け、それぞれセンサSS1, SS2, SS3, SS4, SS5, SS6, SS7, SS8, SS9と名付ける。

【0111】

そして、おのおののセンサSS1, SS2, SS3, SS4, SS5, SS6, SS7, SS8, SS9に含まれる画素のZ値の平均値を、それぞれのセンサSS1, SS2, SS3, SS4, SS5, SS6, SS7, SS8, SS9のZ値として用いる。

【0112】

そして、それぞれのセンサSS1, SS2, SS3, SS4, SS5, SS6, SS7, SS8, SS9には、重み値として「3」、「4」、「3」、「2」、「5」、「2」、「1」、「2」、「1」をそれぞれ割り当てる。

【0113】

この場合、各センサへの重み値の割り当て規則は、「中央が最も大きく、かつ、上から下方向に向かって小さく」というものである。

【0114】

なお、このセンサの個数は、9個に限らず、例えば、16個に設定することができる。

【0115】

図 11, 12 は、Z 値反転判定処理の一例を示している。

【0116】

まず、カラー画像データの全画素の Z 値の平均値 MZ を算出し (処理 301)、変数 S1, S2、および、カウンタ CT の値を 0 に初期設定する (処理 302)。

【0117】

次いで、カウンタ CT の値が 9 より小さいかどうかを調べ (判断 303)、判断 303 の結果が YES になるときには、CT 番目のセンサ SS (SS (CT)) の Z 値が、平均値 MZ 以上であるかどうかを調べる (判断 304)。

【0118】

判断 304 の結果が YES になるときには、変数 S2 に、CT 番目のセンサ S の重み値を積算する (処理 305)。また、判断 304 の結果が NO になるときには、変数 S1 に、CT 番目のセンサ SS の重み値を積算する (処理 306)。

【0119】

これにより、変数 S1 には、画面の明るさの平均値よりも暗いセンサの重み値が加算され、変数 S2 には、画面の明るさの平均値よりも明るいセンサの重み値が加算される。

【0120】

次に、カウンタ CT の値を 1 つ増やし (処理 307)、判断 303 に戻る。

【0121】

判断 303 の結果が NO になるときには、変数 S1 の値が、変数 S2 の値よりも大きいかどうかを調べる (判断 308)。判断 308 の結果が YES になるときには、Z 値を反転する (意味については後述) と判定する (処理 309)。

【0122】

また、判断 308 の結果が NO になるときには、変数 S2 の値が、変数 S1 の値よりも大きいかどうかを調べる (判断 310)。判断 310 の結果が YES になるときには、Z 値を反転しない (意味については後述) と判定する (処理 311)。

【0123】

また、判断310の結果がNOになるときには、変数S1と変数S2の値が等しい場合なので、中央部のセンサSS5のZ値が、平均値MZ以上であるかどうかを調べる（判断312）。

【0124】

判断312の結果がYESになるときには、Z値を反転しないと判定し（処理313）、判断312の結果がYESになるときにはZ値を反転すると判定する（処理314）。

【0125】

次いで、立体視画像表示アプリケーションの表示画面（図示略）の項目のF1/F2モードの設定が、F2モードになっているかどうかを調べる（判断315）。

【0126】

判断315の結果がYESになるときには、そのときの判定結果を反転する（処理316）。すなわち、Z値を反転すると判定している場合には、Z値を反転しないと最終的に決定し、また、Z値を反転しないと判定している場合には、Z値を反転すると最終的に決定する。

【0127】

そして、この処理を終了する。

【0128】

次に、輪郭抽出基礎データ処理（処理106）について説明する。

【0129】

この輪郭抽出基礎データ処理では、図13に示すように、各画素について、当該画素位置を左上隅に配置した縦12ドット（My）で横12ドット（Mx）の二次元マトリクス領域MMを抽出し、その二次元マトリクス領域MMに含まれる144画素のZ値の平均値を算出して、その平均値を、注目画素PxのZ値（outinfo）として割り当てる。なお、この二次元マトリクス領域MMの大きさは、縦12ドットで横12ドットに限ることはなく、適宜な大きさを設定することができる。

【0130】

また、 x 座標が(WD-12)よりも小さい値の注目画素 P_x 、および、 y 座標が(HT-12)よりも小さい値の注目画素 P_x については、二次元マトリクス領域MMに含まれる画素のデータのみを抽出して、平均値を算出し、その算出した平均値を注目画素 P_x のZ値(outinfo)として割り当てる。

【0131】

この輪郭抽出基礎データ処理では、立体視した際の画像の乱れが抑制され、また、画像の輪郭が鮮明になると言う効果が得られる。

【0132】

これらの効果について説明する。まず、立体視するためには、各画素について元の位置から視差に相当する量を左右いずれかに移動する必要がある、その移動量は、Z値から求めている。このために、Z値の変化、すなわち、視差の変化が激しい部分では、画素のクロス（隣接する画素が、それぞれ左右別方向へ移動すること）が発生する。このような画素のクロスは、物体に光が当たってハイライトになっている輪郭の部分などに多く発生し、その結果、輪郭のぼやけ、にじみ、干渉縞模様などが、立体視した画像に出現する。したがって、立体視画像を生成する際には、このような画素のクロス現象が発生しない範囲で視差を決定することが重要である。

【0133】

本実施例の輪郭抽出基礎データ処理では、二次元マトリクス領域MMの範囲で平均値を算出して、注目画素 P_x のZ値として割り当てるようにしているので、極端にZ値が変化している部分のZ値の変化値がなめらかになり、その結果、画素のクロス現象を抑制することができる。

【0134】

すなわち、輪郭抽出基礎データ処理により、画像の乱れが抑制されるとともに、画像の輪郭が鮮明になる。

【0135】

また、画像に含まれる複数物体について、同一物体のZ値の変化を抑制しているので、物体が重なっている部分では、その物体の境界を構成する輪郭が際だつ

こととなり、結果的に、おのこの物体の独立性が保持され、その部分での物体の区別が強調される。

【0136】

図14は、輪郭抽出基礎データ処理（処理106）の一例を示している。

【0137】

ここで、*zinfoは計算されたZ値が格納された領域へのポインタ、WDは元画像の横幅画素数、HTは元画像の縦幅画素数、*outinfoは輪郭抽出基礎データを格納する領域へのポインタをあらわす。

【0138】

まず、カウンタLNを0に初期設定し（処理401）、カウンタLNの値が縦幅画素数HTより小さいかどうかを調べる（判断402）。判断402の結果がNOになるときは、1画面分の処理が終了したので、この処理を終了する。

【0139】

また、判断402の結果がYESになるときには、 $(HT - LN)$ の値が12以上であるかどうかを調べ（判断403）、判断403の結果がYESになるときには、二次元マトリクス領域MMの縦方向のサイズを記憶するための変数MT_hに12を代入し（処理404）、判断403の結果がNOになるときには、変数MT_hに $(HT - LN)$ の値を代入する（処理405）。

【0140】

次に、カウンタCMの値を0に初期設定し（処理406）、カウンタCMの値が横幅画素数WDより小さいかどうかを調べる（判断407）。判断407の結果がYESになるときには、 $(WD - CM)$ の値が12以上であるかどうかを調べ（判断408）、判断408の結果がYESになるときには、二次元マトリクス領域MMの幅方向のサイズを記憶するための変数MT_wに12を代入し（処理409）、判断408の結果がNOになるときには、変数MT_wに $(WD - CM)$ の値を代入する（処理410）。

【0141】

次いで、そのときのカウンタLN、CMの値で示される注目画素について変数MT_h、MT_wで示される二次元マトリクス領域MMに含まれる全画素のZ値（

z i n f o) の合計値 S M 1 を算出し (処理 4 1 1)、注目画素の Z 値として、o u t i n f o に $S M 1 / (M T h * M T w)$ の値 (すなわち、二次元マトリクス領域 M M の Z 値の平均値) を格納する (処理 4 1 2)。

【0142】

次いで、カウンタ C M の値を 1 つ増やして (処理 4 1 3)、判断 4 0 7 へ戻る。

【0143】

また、判断 4 0 7 の結果が N O になるとときには、1 ライン分の画素についての処理が終了したので、カウンタ L N の値を 1 つ増やして (処理 4 1 4)、判断 4 0 2 へ戻り、それ以降の処理を実行する。

【0144】

次に、Z 値拡張処理 (処理 1 0 7) について説明する。

【0145】

まず、輪郭抽出基礎データ処理で抽出された各画素の Z 値は、二次元マトリクス領域 M M における平均値として算出しているため、画像全体の平均的な Z 値方向に引き下げられたり、あるいは、引き上げられたりする傾向が現れる (図 1 6 参照)。

【0146】

そこで、図 1 5 に一点鎖線で示したように、Z 値の画素分布が広がるように、Z 値を修正する。これが、Z 値拡張処理である。

【0147】

図 1 6, 1 7 は、Z 値拡張処理の一例を示している。

【0148】

ここで、* z _ i m a g e はグレースケール Z 値を格納した領域をあらわすポインタ、a v e _ z は画像の平均 Z 値、m a x _ z は画像最大 Z 値、m i n _ z は画像最小 Z 値、W D は元画像の横幅画素数、H T は元画像の縦幅画素数、* a d j u s t _ z _ i m a g e は修正 Z 値を格納する領域をあらわすポインタである。

【0149】

まず、1画面中の画素について、 \max_z 、 \min_z 、 $\text{ave_}z$ をそれぞれ算出する(処理501)。次いで、変数 \max_span に $(\max_z - \text{ave_}z)$ の値を代入し、変数 \min_span に $(\min_z - \text{ave_}z)$ の値を代入する(処理502)。

【0150】

次に、色彩深度が16ビットであるかどうかを調べて(判断503)、判断503の結果がYESになるとときには、変数 \max に16を代入し(処理504)、判断503の結果がNOになるとときには(色彩深度が24ビットまたは32ビットの場合)、変数 \max に255を代入する(処理505)。

【0151】

次いで、カウンタ L_n を0に初期設定し、 pz_image に z_image を代入する(処理506)。

【0152】

そして、カウンタ L_n が縦幅画素数 HT よりも小さいかどうかを調べる(判断507)。判断507の結果がNOになるとときには、全ての画素について処理が終了した場合なので、この処理を終了する。

【0153】

判断507の結果がYESになるとときには、カウンタ $clmn$ を0に初期設定し(処理508)、カウンタ $clmn$ の値が横幅画素数 WD よりも小さいかどうかを調べる(判断509)。判断509の結果がNOになるとときには、1ライン分の画素の処理が終了した場合なので、カウンタ L_n の値を1つ増やして(処理510)、判断507へ戻る。

【0154】

また、判断509の結果がYESになるとときには、 $*pz_image$ の値が画素平均値 $\text{ave_}z$ よりも大きいかどうかを調べる(判断511)。判断511の結果がYESになるとときには、変数 z に、 $(z - \text{ave_}z) / \max_span * (\max - \text{ave_}z)$ の値を代入する(処理512)。

【0155】

また、判断511の結果がNOになるとときには、 $*pz_image$ の値が画

素平均値 ave_z よりも小さいかどうかを調べる (判断 513)。判断 513 の結果が YES になるときには、変数 z に、 $(z - ave_z) / (min_z - ave_z) * min_span$ の値を代入する (処理 514)。また、判断 513 の結果が NO になるとき、すなわち、 $*pz_image$ の値が画素平均値 ave_z に等しいときには、変数 z に 0 を代入する (処理 515)。

【0156】

このようにして、変数 z に値の代入が終了すると、 $*adjust_z_image$ に、 $*pz_image + z$ の値を代入し (処理 516)、注目画素について、 Z 値の修正値を格納する。

【0157】

次に、 pz_image の値を 1 つ増やし、 $adjust_z_image$ の値を 1 つ増やし、 $clmn$ の値を 1 つ増やして (処理 517)、判断 509 へ戻り、次の画素について Z 値の修正値を算出、格納する。

【0158】

次に、参照 Z 値算出処理 (処理 108) について説明する。

【0159】

まず、この参照 Z 値算出処理は、 Z 値拡張処理で算出した Z 値を修正するために参照する値を算出する処理である。

【0160】

この参照 Z 値算出処理では、図 18 に示すように、カラー画像データによりあらわされる画面 GG に含まれる 1 つ以上の物体 $OB1$ 、 $OB2$ を抽出し、今までの算出方法とは異なる手法により、それぞれの物体 $OB1$ 、 $OB2$ について、含まれる 1 つ以上の Z 値を算出し、最終的には、修正 Z 値を算出する。

【0161】

すなわち、参照 Z 値算出処理は、図 19 に示すように、カラー画像データのそれぞれの画素について所定の丸め係数で除算して丸め奥行き値を算出し、その丸め奥行き値が同一値となる画素を結ぶ等高線 (図 18 の等高線 $HL1$ 、 $HL2$ 、 $HL3$ 、 $HL4$ 、 $HL5$) を 1 つ以上作成する等高線数値化処理 (処理 701 (=等高線数値化処理ステップ)) と、その作成した等高線に基づいて、カラー画

像データに含まれる1つ以上の独立物体（図18の物体OB1, OB2）を抽出する独立物体特定処理（処理702（＝独立物体特定処理ステップ））と、その抽出した独立物体の重心（図18の点PPc）を算出し、その重心からカラー画像データの画面GGの下辺に垂直に下ろした垂線（図18の線VL）と、当該独立物体に含まれる1つ以上の等高線の交点（図18の点PP1, PP2, PP3, PP4, PP5）を算出する交点算出処理（処理704（＝交点算出処理ステップ））と、算出したおのものの交点について、重心のy座標、当該交点のy座標、および、あらかじめ設定されている距離情報に基づいて、第2の奥行き値を算出する新Z値算出処理（処理704（＝新奥行き値算出処理ステップ））と、丸め奥行き値と新Z値の差を奥行き値修正値として算出する修正Z値算出処理（処理705（＝修正奥行き値算出処理ステップ））からなる。

【0162】

また、独立物体の最も内側の等高線HL1で囲まれる部分、および、隣り合う2つの等高線で囲まれる部分を、等高面と称し、各等高面に含まれる画素には、等高線HL1に割り当てられるZ値、または、2つの等高線のうち外側の等高線に割り当てられるZ値が、それぞれ新Z値として割り当てられる。

【0163】

図20, 21, 22は、等高線数値化処理（処理701）の一例を示している。

【0164】

ここで、*outlineは輪郭抽出基礎データの格納領域をあらわすポインタ、WDは元画像の横幅画素数、HTは元画像の縦幅画素数、*cordはZ値ベクトル格納用領域をあらわすポインタである。また、cordは、Vinfoという構造体のデータであり、Vinfoという構造体は、Z（グレースケール値）、npoint（整数）、および、*point（座標データ）というデータ要素を持つ。

【0165】

まず、定数maxに、255を丸め係数（後述）で除算した結果を代入し、変数Target_zに0を代入する（処理710）。次に、変数Target_

z の値が定数maxよりも大きくなったかどうかを調べる（判断711）。判断711の結果がNOになるとときには、この処理を終了する。

【0166】

また、判断711の結果がYESになるとときには、カウンタLnを0に初期設定し（処理712）、カウンタLnの値がHTよりも小さいかどうかを調べる（判断713）。判断713の結果がNOになるとときには、Target_zの値を1つ増やして（処理714）、判断711へ戻る。

【0167】

判断713の結果がYESになるとときには、カウンタCLMに0を代入し、変数P1に-1を代入し、変数P2に-1を代入し、変数fndに0を代入する（処理715）。

【0168】

カウンタCLMの値がWDよりも小さいかどうかを調べ（判断716）、判断716の結果がYESになるとときには、*outlineの値が、Target_zの値に等しいかどうかを調べる（判断717）。判断717の結果がYESになるとときには、変数P1の値が0以上であるかどうかを調べ（判断718）、判断718の結果がNOになるとときには、変数P1にカウンタCLMの値を代入し（処理719）、カウンタCLMの値を1つ増やして（処理720）、判断716へ戻る。また、判断718の結果がYESになるとときには、処理720へ移行し、それ以降の処理を実行する。

【0169】

また、判断717の結果がNOになるとときには、変数P1の値が0以上であるかどうかを調べ（判断721）、判断721の結果がYESになるとときには、変数P2にカウンタCLMの値を代入し、変数fndの値を1つ増やし（処理722）、処理720へ移行し、それ以降の処理を実行する。また、判断721の結果がNOになるとときには、処理720へ移行し、それ以降の処理を実行する。

【0170】

また、判断716の結果がNOになるとときには、変数fndの値が0よりも大きいかどうかを調べ（判断723）、判断723の結果がNOになるとときには、

カウンタ L_n の値を 1 つ増やして (処理 724)、判断 713 へ戻る。

【0171】

また、判断 723 の結果が YES になるときには、当該画素位置について、Z 値が変化している画素を探索する。この Z 値変化画素探索順序は、図 21 (b) に示した順序に従う。すなわち、注目画素 P1 の直上位置にある画素の Z 値が変化しているかどうかを調べ (判断 730)、判断 730 の結果が NO になるときには、注目画素 P1 の右上位置にある画素の Z 値が変化しているかどうかを調べ (判断 731)、判断 731 の結果が NO になるときには、注目画素 P1 の左上位置にある画素の Z 値が変化しているかどうかを調べ (判断 732)、判断 732 の結果が NO になるときには、注目画素 P1 の右下位置にある画素の Z 値が変化しているかどうかを調べ (判断 733)、判断 733 の結果が NO になるときには、注目画素 P1 の直下位置にある画素の Z 値が変化しているかどうかを調べ (判断 734)、判断 734 の結果が NO になるときには、注目画素 P1 の左下位置にある画素の Z 値が変化しているかどうかを調べ (判断 735)、判断 735 の結果が NO になるときには、エラー終了する。

【0172】

また、判断 730、731、732、733、734、735 の結果が YES になるときには、見つけた画素の座標が既に発見済の画素であるかどうかを調べて、ベクトルを一周したかどうかを調べる (判断 736)。判断 736 の結果が YES になるときには、この処理を終了する。

【0173】

また、判断 736 の結果が NO になるときには、cord の point の x と、cord の point の y の値に、それぞれ発見した画素の座標を保存する (処理 737)。また、cord の point の値を 1 つ増やし (処理 738)、cord の npoint の値を 1 つ増やし (処理 739)、注目画素 P1 を発見位置に移動して (処理 740)、判断 730 へ移動し、次のベクトルについて処理を行う。

【0174】

図 23、24、25 は、独立物体特定処理 (処理 702) の一例を示している

【0175】

ここで、*cordはZ値ベクトル格納用領域をあらわすポインタ、*cent__cordは求める中心矩形を格納するベクトル領域をあらわすポインタ、centerは求める中心点の座標データ、*side__cordは求める周縁矩形を格納するベクトル領域をあらわすポインタである。

【0176】

まず、処理対象となるZ値ベクトルを1つ選択する(処理750)。次に、g__min__xに999999を代入し、g__max__xに0を代入し、g__min__yに999999を代入し、g__max__yに0を代入して、それらの値を初期設定する(処理751)。また、cordpにcordの値を代入する(処理752)。

【0177】

次に、cordpのnpointの値が0よりも大きいかどうかを調べ(判断753)、判断753の結果がNOになるときには、全てのベクトルについての処理が終了したかどうかを調べ(判断754)、判断754の結果がNOになるときには、処理750へ戻り、次のZ値ベクトルについての処理を行う。また、判断754の結果がYESになるときには、このときの処理を終了する。

【0178】

判断753の結果がYESになるときには、カウンタcntの値を0に初期設定し(処理755)、min__xに999999を代入し、max__xに0を代入し、min__yに999999を代入し、max__yに0を代入して、それらの値を初期設定し、変数pntにcordpのpointの値を代入する(処理756)。

【0179】

次いで、カウンタcntの値がcordpのnpointよりも小さいかどうかを調べ(判断757)、判断757の結果がYESになるときには、変数pntのxの値が、min__xよりも小さいかどうかを調べる(判断758)。判断758の結果がYESになるときには、水平最小位置をp1に代入する。すなわ

ち、p 1. x の値に p n t の x の値を代入し、p 1. y の値に p n t の y の値を代入し、m i n__x の値に p 1. x の値を代入する（処理 7 5 9）。また、判断 7 5 8 の結果が N O になるとときには、処理 7 5 9 は実行しない。

【0180】

次いで、p n t の x の値が m a x__x よりも大きいかどうかを調べる（判断 7 6 0）。判断 7 6 0 の結果が Y E S になるとときには、水平最大位置を p 2 に代入する。すなわち、p 2. x の値に p n t の x の値を代入し、p 2. y の値に p n t の y の値を代入し、m a x__x の値に p 2. x の値を代入する（処理 7 6 1）。また、判断 7 6 0 の結果が N O になるとときには、処理 7 6 1 は実行しない。

【0181】

次に、変数 p n t の y の値が、m i n__y よりも小さいかどうかを調べる（判断 7 6 2）。判断 7 6 2 の結果が Y E S になるとときには、垂直最小位置を p 3 に代入する。すなわち、p 3. x の値に p n t の x の値を代入し、p 3. y の値に p n t の y の値を代入し、m i n__y の値に p 3. y の値を代入する（処理 7 6 3）。また、判断 7 6 2 の結果が N O になるとときには、処理 7 6 3 は実行しない。

【0182】

次いで、p n t の y の値が m a x__y よりも大きいかどうかを調べる（判断 7 6 4）。判断 7 6 4 の結果が Y E S になるとときには、垂直最大位置を p 4 に代入する。すなわち、p 4. x の値に p n t の x の値を代入し、p 4. y の値に p n t の y の値を代入し、m a x__y の値に p 2. y の値を代入する（処理 7 6 5）。また、判断 7 6 4 の結果が N O になるとときには、処理 7 6 5 は実行しない。

【0183】

そして、カウンタ c n t の値を 1 つ増やし、p n t の値を 1 つ増やして（処理 7 6 6）、判断 7 5 7 へ戻る。

【0184】

また、判断 7 5 7 の結果が N O になるとときには、p 1. x の値が g__m i n__x の値よりも小さいかどうかを調べ（判断 7 6 7）、判断 7 6 7 の結果が Y E S になるとときには、c o r d p を c e n t__c o r d に複写し（処理 7 6 8）、g

__min__xにp1. xの値を代入する(処理769)

【0185】

次いで、点(p1. x、p3. y)と点(p2. x、p4. y)を結ぶ線分と、点(p2. x、p3. y)と点(p1. x、p4. y)を結ぶ線分の交点を求め、その交点(重心PPc)のx座標とy座標の値を、それぞれcenter. x、center. yに保存する(処理770)。

【0186】

また、判断767の結果がNOになるときには、処理768～770を実行しない。

【0187】

次に、p2. xがg__max__xよりも大きいかどうかを調べ(判断771)、判断771の結果がYESになるときには、cordpをside__cordに複写し(処理772)、g__max__xにp2. xを代入する(処理773)。また、判断771の結果がNOになるときには、処理772、773を実行しない。

【0188】

そして、cordpの値を1つ増やし(処理774)、判断753へ戻る。

【0189】

図26は、交点算出処理(処理703)の一例を示している。

【0190】

ここで、*cordはZ値ベクトル格納用領域をあらわすポインタ、*cent__cordは求める中心矩形を格納するベクトル領域をあらわすポインタ、centerは求める中心点の座標データ、*side__cordは求める周縁矩形を格納するベクトル領域をあらわすポインタ、p[]は、交点の座標データを格納する配列である。

【0191】

まず、Z値ベクトルを1つ選択し(処理780)、cordpにcordを代入し、fndを0に初期設定する(処理781)。次いで、cordpのnpointの値が0よりも大きいかどうかを調べ(判断782)、判断782の結果

がNOになるときには、全ベクトルの処理が終了したかどうかを調べ（判断783）、判断783の結果がNOになるときには、処理780へ戻り、次のZ値ベクトルについて処理を実行する。また、判断783の結果がYESになるときには、この処理を終了する。

【0192】

判断782の結果がYESになるときには、pntにcordpのpointの値を代入し、cntの値を0に初期設定する（処理784）。

【0193】

次に、cntが、cordpのnpointよりも小さいかどうかを調べ（判断785）、判断785の結果がYESになるときには、pnt.xとcenter.xが等しいかどうかを調べる（判断786）。

【0194】

判断786の結果がYESになるときには、pnt.yがcenter.yよりも大きいかどうかを調べる（判断787）。判断787の結果がYESになるときには、中心点から垂直に下ろした垂線と等高線との交点を求める。すなわち、p[find].xにpnt.xの値を代入し、p[find].yにpnt.yの値を代入する（処理788）。

【0195】

次いで、findの値を1つ増やし（処理789）、また、データの終端を示すためにp[find].xとp[find].yにそれぞれ-1を代入する（処理790）。次に、cordpの値を1つ増やして（処理791）、判断782へ戻る。

【0196】

また、判断786の結果がNOになるとき、および、判断787の結果がNOになるときには、cntの値を1つ増やし、pntの値を1つ増やして（処理792）、判断785へ戻る。

【0197】

また、判断785の結果がNOになるときには、処理791へ移行する。

【0198】

図 27 は、新 Z 値算出処理（処理 704）の一例を示している。

【0199】

ここで、`center` は求めた中心点の座標データ、`p []` は、求めた交点の座標データを格納する配列、`dist` は、表示しているカラー画像データの画面をカメラで撮影したと設定したときのカメラから表示画面までの距離（定数）、`focus` は上記カメラの焦点距離（定数）、`HT` は元画像の縦幅画素数、`new_z []` は、新 Z 値を格納する配列である。

【0200】

まず、処理対象を 1 つ選択し（処理 795）、`cnt` の値を 0 に初期設定し（処理 796）、`cent_y` に `HT` から `center.y` を減じた値を代入し、`object_h` に、 $(dist / focus * center_h)$ の値を代入する（処理 797）。

【0201】

次いで、`p[cnt].x` の値が 0 以上であるかどうかを調べ（判断 798）、判断 798 の結果が YES になるときには、`new_z[cnt]` に、 $(object_h * focus / (p[cnt].y - center.y) - dist)$ の値を代入する（処理 799）。

【0202】

そして、`cnt` の値を 1 つ増やして（処理 800）、判断 798 へ戻る。

【0203】

また、判断 798 の結果が NO になるときには、全ての対象について処理が終了したかどうかを調べ（判断 801）、判断 801 の結果が NO になるときには、処理 795 へ戻り、他の対象に対しての処理を行う。また、判断 801 の結果が YES になるときには、この処理を終了する。

【0204】

図 28 は、修正 Z 値算出処理（処理 705）の一例を示している。

【0205】

ここで、`p []` は求めた交点を格納する配列、`new_z []` は新 Z 値を格納する配列、`adjust_outline []` は修正分 Z 値を格納する配列であ

る。

【0206】

まず、処理対象を1つ選択し（処理805）、`cnt`の値に1を代入する（処理806）。

【0207】

次いで、`p[cnt].x`の値が0以上であるかどうかを調べる（判断807）。判断807の結果がNOになるときには、1つの処理対象についての処理が終了したので、全ての対象に対する処理を終了したかどうかを調べる（判断808）。判断808の結果がNOになるときには、処理805へ戻り、他の対象についての処理を行う。また、判断808の結果がYESになるときには、この処理を終了する。

【0208】

判断807の結果がYESになるときには、`oz1`に（`p[cnt-1].x`、`p[cnt-1].y-1`）の座標のZ値を代入し（処理809）、`oz2`に（`p[cnt].x`、`p[cnt].y-1`）の座標のZ値を代入する（処理810）。

【0209】

次いで、`sub1`に（`oz2-oz1`）の値を代入し（処理811）、`sub2`に（`new_z[cnt]-new_z[cnt-1]`）の値を代入する（処理812）。

【0210】

そして、`sub1`と`sub2`の符号が同一であるかどうかを調べ（判断813）、判断813の結果がYESになるときには、`adjust_outline[cnt]`に0を代入し（処理815）、判断813の結果がNOになるときには、`adjust_outline[cnt]`に（`sub2-sub1`）の値を代入する（処理815）。

【0211】

次いで、`cnt`の値を1つ増やして（処理816）、判断807へ戻る。

【0212】

図29は、Z値修正処理（処理109）の一例を示している。このZ値修正処理では、上述した等高面に含まれる全ての画素について、同一の修正Z値が割り当てられる。

【0213】

ここで、*z__imageはグレースケールZ値を格納している領域をあらわすポインタ、WDは元画像の横幅画素数、HTは元画像の縦幅画素数、*adjust__z__imageは修正Z値を格納している領域をあらわすポインタである。

【0214】

まず、Lnに0を代入し（処理901）、Lnの値がHTより小さいかどうかを調べる（判断902）。判断902の結果がNOになるときには、この処理を終了する。

【0215】

判断902の結果がYESになるときには、clmnに0を代入して（処理903）、clmnがWDよりも小さいかどうかを調べる（判断904）。判断904の結果がYESになるときには、そのときの注目画素に対応するadjust__z__imageを取得し（処理905）、z__imageを、z__imageとadjust__z__imageを加算した値で更新する（処理906）。

【0216】

次いで、clmnの値を1つ増やし（処理907）、判断904へ戻る。判断904の結果がNOになるときには、Lnの値を1つ増やして（処理908）、判断902へ戻り、それ以降の処理を行う。

【0217】

図30は、立体変換出力処理（処理110）の一例を示している。

【0218】

ここで、*srcは元画像の格納領域をあらわすポインタ、*outinfoは輪郭抽出基礎データを格納した領域をあらわすポインタ、WDは元画像の横幅画素数、HTは元画像の縦幅画素数、*image3dは立体視画像データを格納する領域をあらわすポインタである。

【0219】

まず、LNを0に初期設定し（処理1001）、LNがHTより小さいかどうかを調べる（判断1002）。判断1002の結果がNOになるとときには、この処理を終了する。また、1002の結果がYESになるとときには、CNを0に初期設定する（処理1003）、。

【0220】

次いで、CNがWDよりも小さいかどうかを調べ（判断1004）、判断1004の結果がYESになるとときには、そのときのoutinfoに対応した適用Z値（ZZ）を算出する適用Z値算出処理（処理1005）を行い、続いて、適用Z値に基づいて視差値（SS）を算出する視差値算出処理（処理1006）を行う。

【0221】

このようにして、視差SSが算出されると、LNの値が偶数であるかどうかを調べ（判断1007）、判断1007の結果がYESになるとときには、MVに（CN+SS）の値を代入し（処理1008）、また、判断1007の結果がNOになるとときには、MVに（CN-SS）の値を代入する（処理1009）。

【0222】

次いで、元画像データの（src+MV）の位置の画素のデータをimage3dへ格納する（処理1010）。

【0223】

次に、image3dの値を1つ増やし、outinfoの値を1つ増やし、CNの値を1つ増やして（処理1011）、判断1004へ戻る。

【0224】

また、判断1004の結果がNOになるとときには、srcを次の行頭へ進め（処理1012）、LNの値を1つ増やして（処理1013）、判断1002へ戻り、それ以降の処理を行う。

【0225】

図31は、適用Z値算出処理（処理1005）の一例を示している。

【0226】

まず、256をオブジェクト深度で除算して、その値を丸め係数MKに代入し（処理1020）、`z_image`の値をMKで除算した結果を、ZZに代入する（処理1021）。なお、ここで算出した丸め係数MKは、処理710で用いる丸め係数と同じものである。すなわち、処理710における丸め係数は、この丸め係数MKを用いる。

【0227】

ここで、ZZの値が上限基準値よりも大きいかどうかを調べ（判断1022）、判断1022の結果がYESになるときには、ZZに上限基準値を代入する（処理1023）。また、判断1022の結果がNOになるときには、処理1023を実行しない。

【0228】

次いで、ZZの値が下限基準値よりも小さいかどうかを調べ（判断1024）、判断1024の結果がYESになるときには、ZZに下限基準値を代入する（処理1025）。また、判断1024の結果がNOになるときには、処理1025を実行しない。

【0229】

なお、オブジェクト深度、上限基準値、下限基準値、および、奥行深度は、例えば、立体視画像表示アプリケーションにおいて表示パラメータをユーザが適宜に入力できるようにするための設定ダイアログパネル（図示略）等において、ユーザが任意に入力できるようにすることができる。

【0230】

図32は、視差値算出処理（処理1006）の一例を示している。

【0231】

まず、処理105のZ値反転判定処理において、Z値を反転する旨が判定出力されているかどうかを調べる（判断1030）。

【0232】

判断1030の結果がYESになるときには、ZZに $(256 - ZZ)$ の値を代入することで、ZZの値を修正する（処理1031）。次いで、ZZに奥行深度を加算して、その結果を視差値SSに代入して（処理1032）、この処理を

終了する。

【0233】

また、判断1030の結果がNOになるとときには、処理1032へ移行し、Zに奥行深度を加算した値を視差値SSに代入して、この処理を終了する。

【0234】

このようにして、本実施例では、Z値算出処理により、Z値の元になるグレースケール値として、人間の網膜に存在する赤錐体、緑錐体、および、青錐体の数の分布に基づき、立体視画像を作成する際に特に適する重み付けの値を与えたRデータの30%と、Gデータの59%と、Bデータの11%の総和を取った値を用いているので、人間がより自然と感じる奥行き値を算出することができる。

【0235】

また、Z値反転判断処理により、白い部分を遠くにするか、あるいは、黒い部分を遠くにするかを判断しており、それにより、不自然な立体感を除去し、ごく自然な立体感を実現することができる。

【0236】

また、参照Z値算出処理により、画像データに含まれる1つ以上の独立物体を抽出し、その独立物体の周縁部を結ぶ等高線をおのおのの奥行き値について形成するとともに、同一奥行き値を有する等高面を決定し、さらに、その等高面の奥行き値を、あらかじめ設定した距離条件に従って補正しているので、奥行き値の順序性を正しいものとして修正することができ、それにより、不自然な立体感を除去し、ごく自然な立体感を実現することができる。

【0237】

また、輪郭抽出基礎データ処理により、二次元マトリクス領域MMの範囲で平均値を算出して、注目画素P_xのZ値として割り当てるようにしているので、極端にZ値が変化している部分のZ値の変化値がなめらかになり、その結果、画素のクロス現象を抑制することができ、画像の乱れが抑制される。また、物体が重なっている部分では、その物体の境界を構成する輪郭が際だつこととなり、結果的に、おのおのの物体の独立性が保持され、その部分での物体の区別が強調される。

【0238】

また、Z値が高い画素やZ値が低い画素の数は、画像データ全体の画素数に比べて非常に少ない数であるから、上限基準値や下限基準値の処理で特定値を超えるZ値を一定範囲内に収まるように再配置しているので、一定範囲内に納めたZ値を画素クロスが発生しない限度一杯に拡張することで、立体感を損なわずに画像のゆがみを抑えることができることとなった。

【0239】

また、ユーザが立体視画像表示アプリケーションの表示画面で適宜に設定した入力値、および、設定ダイアログパネルで適宜に入力した入力値に従って、立体視画像データの視差値が算出されるので、ユーザは、所望する態様の立体視画像を適宜に表示させることができる。

【0240】

また、上述した処理は、動画の1フレームについて行われるので、ユーザが上述した入力値を適宜に変更すると、その変更を反映した立体視画像または非立体視画像が、次のフレームから表示されるので、非常に便利であり、使い勝手が良好である。

【0241】

図33は、本発明の他の実施例にかかる立体視画像表示装置の一例を示している。

【0242】

同図において、CPU（中央処理装置）21は、この立体視画像表示装置の動作制御、立体視用画像データ作成処理などの各種処理を実行するためのものであり、ROM（リード・オンリ・メモリ）22は、起動時などに実行されるプログラムや参照データなどが記憶されるものであり、RAM（ランダム・アクセス・メモリ）23は、CPU1のワークエリアなどを実現するためのものであり、キャラクタジェネレータ24は、図形文字の表示データ（フォントデータ）を発生するものであり、時計回路25は現在時刻情報を出力するためのものである。

【0243】

インターネットインターフェース回路26は、この立体視画像表示装置をイン

ターネット（図示略）へ接続するためのものであり、インターネット伝送制御部 27 は、インターネットを介して、他の端末装置との間で種々のデータをやりとりするための各種所定のプロトコルスイートの通信制御処理を実行するためのものである。

【0244】

磁気ディスク装置 28 は、この立体視画像表示装置で実行される各種処理やオペレーティングシステムなどのプログラムデータを記憶するとともに、各種処理により作成される各種データなどを記憶する大容量の外部記憶装置であり、CD-ROM 装置 29 は、CD-ROM 30 に記憶されているデータを再生するためのものである。

【0245】

キーボード装置 31 は、フルキー配列を備えて、ユーザがキー操作を行うためのものであり、画面指示装置 32 は、ユーザが表示画面をクリック操作するなどの種々の操作を行うためのものであり、入力制御部 33 は、キーボード装置 31 および画面指示装置 32 の入力データを取り込むためのものである。

【0246】

表示制御部 34 は、立体視表示モードと非立体視モードの表示モードを備えたものであり、立体視表示モード時には、左目画像データバッファ 35 に左目画像データを保存し、また、右目画像データバッファ 36 に右目画像データを保存し、切換器 37 を介して、左目画像データバッファ 35 または右目画像データバッファ 36 を、画面表示装置 38 の表示サイクルに同期して交互に切換選択し、液晶表示装置 38 へ接続する。また、非立体視モードの場合には、左目画像データバッファ 35 に画像データを保存し、切換器 37 により左目画像データバッファ 35 を選択して、画面表示装置 38 へ接続する。

【0247】

また、表示制御部 34 は、立体視表示モードが設定されている場合には、切換器 37 の切換動作に同期して、液晶シャッタメガネ 39 の動作を切換える。すなわち、切換器 37 が左目画像バッファ 35 を選択している状態では、液晶シャッタメガネ 39 の左目をオン（透過）状態に設定するとともに右目をオフ（遮断）

状態に設定し、切換器 37 が右目画像バッファ 36 を選択している状態では、液晶シャッタメガネ 39 の右目をオン（透過）状態に設定するとともに左目をオフ（遮断）状態に設定する。

【0248】

ここで、画面表示装置 38 は、プラズマパネルディスプレイ装置や、液晶表示装置のように、1 画面単位に画像を表示切り替えするものである。すなわち、CRT 表示装置のように、ライン単位に画像を表示するものとは異なる表示機構を備えたものである。

【0249】

これらの CPU 21、ROM 22、RAM 23、キャラクタジェネレータ 24、時計回路 25、インターネット伝送制御部 27、磁気ディスク装置 28、CD-ROM 装置 29、入力制御部 33、および、表示制御部 34 は、内部バス 40 に接続されており、これらの各種要素間のデータのやりとりは、主としてこの内部バス 40 を介して行われている。

【0250】

図 34 は、この立体視画像表示装置で、動画を表示する場合の処理の一例を示している。

【0251】

まず、表示制御部 34 を非立体視モード（2D モード）に設定し（処理 S301）、動画データの 1 フレーム分の画像データを左目画像データバッファ 35 へ出力する（処理 S302）。

【0252】

次いで、表示中の動画データの次のフレームがあるかどうかを調べ（判断 S303）、判断 S303 の結果が NO になるときには、表示終了して（処理 S304）、この処理を終了する。

【0253】

また、判断 S303 の結果が YES になるときには、ユーザが 2D モード（非立体視モード）と 3D モード（立体視表示モード）を切換操作するか、あるいは、画像表示の終了が指定されるまで待つ（判断 S305, S306）。

【0254】

画像表示の終了が指定された場合で、判断S306の結果がYESになるときは、処理S304へ移行し、画像表示を終了して、この表示動作を終了する。また、判断S306の結果がNOになるときは、処理S302へ戻り、次のフレームの表示動作を行う。

【0255】

これにより、初期状態では、立体視画像表示装置のユーザは、非立体視モードの動画を見ることができる。

【0256】

また、ユーザが2Dモード（非立体視モード）と3Dモード（立体視表示モード）を切替操作した場合で、判断S305の結果がYESになるときは、表示制御部34に立体視表示モードをセットし（処理S307）、動画データの1フレーム分の画像データを画像バッファへ取り込み（処理S308）、その1フレーム分の画像データについて所定の立体視表示データ作成処理を行い（処理S309）、それによって得た立体視表示データのうち、左目画像データを左目画像データバッファ35に保存し、右目画像データを右目画像データバッファ36に保存する（処理S310）。

【0257】

これにより、ユーザは、立体視表示モードの動画を見ることができる。

【0258】

また、この状態で、表示中の動画データの次のフレームがあるかどうかを調べ（判断S311）、判断S311の結果がNOになるときは、処理S304へ移行し、表示終了して、この処理を終了する。

【0259】

また、判断S311の結果がYESになるときは、ユーザが2Dモード（非立体視モード）と3Dモード（立体視表示モード）を切替操作するか、あるいは、画像表示の終了が指定されるのを調べる（判断S312, S313）。判断S313の結果がNOになるときは、処理S308へ戻り、立体視表示モードの動画データの表示動作を継続する。

【0260】

また、表示終了が指令された場合で、判断S313の結果がYESになるときには、処理S304へ移行し、表示終了して、この処理を終了する。

【0261】

また、ユーザが2Dモード（非立体視モード）と3Dモード（立体視表示モード）を切換操作した場合で、判断S312の結果がYESになるときには、処理S301へ戻り、表示制御部34を非立体視モード（2Dモード）に設定して、それ以降の処理を実行する。

【0262】

このようにして、本実施例では、1つの画面表示装置38を用いて、立体視表示モードの動画を表示することができるので、立体視表示装置の利用形態を拡大することができる。

【0263】

ところで、立体視表示装置を利用して、運転シミュレータなどの表示画面を、立体視表示させると、非常に有効な訓練を行うことができる。

【0264】

その場合の運転シミュレータ装置の一例を図35に示す。

【0265】

同図において、操作ユニット41は、運転シミュレータ装置が模擬している対象の操作を適切に行うことができるように構成されたものであり、その操作情報は、シミュレーション処理部42に加えられる。

【0266】

シミュレーション処理部42は、操作ユニット41から入力される操作情報に基づき、所定のシミュレーション処理を実行し、ユーザに表示提供する画面の表示データを作成し、その表示データをフレームバッファ43へ出力する。

【0267】

フレームバッファ43のデータは、立体視表示装置44に加えられている。立体視表示装置44は、操作ユニット41から2Dモード（非立体視表示モード）が指定されている場合には、非立体視表示モードの画面を表示し、また、操作ユ

ニット 41 から 3D モード（立体視表示モード）が指定されている場合には、フレームバッファ 43 に保存されている 1 画面分の表示データに基づいて、立体視表示データを作成し、立体視画像を表示する。

【0268】

また、立体視表示装置の応用分野としては、通信端末装置、デジタルスチールカメラ装置、受像器、ストレージ型表示装置、内視鏡等の遠隔医療器具等がある。

【0269】

【発明の効果】

以上説明したように、本発明によれば、立体視画像表示機能を備えた画像表示装置において、非立体視画像表示モードと立体視表示モードの切換スイッチと、立体視画像データを作成する立体視画像データ作成手段を備え、上記切換スイッチにより非立体視画像表示モードが選択されたときには、表示画像データを非立体視画像表示モードで表示する一方、上記切換スイッチにより立体視画像表示モードが選択されたときには、表示画像データを上記立体視画像データ作成手段により立体視画像データに変換し、その立体視画像データを立体視表示モードで表示するようにしたので、ユーザは、適宜に立体視画像と非立体視画像を切り換えて表示させることができ、非常に便利である。

【図面の簡単な説明】

【図 1】

本発明の一実施例にかかる移動体通信端末装置を示したブロック図。

【図 2】

移動体通信端末装置の筐体の概略構成を示した正面図。

【図 3】

移動体通信端末装置で、静止画を表示する場合の処理の一例を示したフローチャート。

【図 4】

移動体通信端末装置で、動画を受信して表示する場合の処理の一例を示したフローチャート。

【図 5】

立体視表示データ作成処理（処理 S107, S211）の一例を示したフローチャート。

【図 6】

カラー画像データの 1 画素分のデータ構造を説明するための概略図。

【図 7】

Z 値算出処理（処理 104）の一例を示したフローチャート。

【図 8】

Z 値算出処理（処理 104）の一例を示したフローチャート（続き）。

【図 9】

Z 値算出処理（処理 104）の一例を示したフローチャート（続き）。

【図 10】

Z 値判定反転処理で適用するセンサの配置例を示した概略図。

【図 11】

Z 値反転判定処理の一例を示したフローチャート。

【図 12】

Z 値反転判定処理の一例を示したフローチャート（続き）。

【図 13】

輪郭抽出基礎データ処理において画素値を算出するための二次元マトリクス領域 MM を説明するための概略図。

【図 14】

輪郭抽出基礎データ処理（処理 106）の一例を示したフローチャート。

【図 15】

Z 値拡張処理を説明するためのグラフ図。

【図 16】

Z 値拡張処理の一例を示したフローチャート。

【図 17】

Z 値拡張処理の一例を示したフローチャート（続き）。

【図 18】

参照 Z 値算出処理で適用するモデルを説明するための概略図。

【図 19】

参照 Z 値算出処理の一例を示したフローチャート。

【図 20】

等高線数値化処理（処理 701）の一例を示したフローチャート。

【図 21】

等高線数値化処理（処理 701）の一例を示したフローチャート（続き）。

【図 22】

等高線数値化処理（処理 701）の一例を示したフローチャート（続き）。

【図 23】

独立物体特定処理（処理 702）の一例を示したフローチャート。

【図 24】

独立物体特定処理（処理 702）の一例を示したフローチャート（続き）。

【図 25】

独立物体特定処理（処理 702）の一例を示したフローチャート（続き）。

【図 26】

交点算出処理（処理 703）の一例を示したフローチャート。

【図 27】

新 Z 値算出処理（処理 704）の一例を示したフローチャート。

【図 28】

修正 Z 値算出処理（処理 705）の一例を示したフローチャート。

【図 29】

Z 値修正処理（処理 109）の一例を示したフローチャート。

【図 30】

立体変換出力処理（処理 110）の一例を示したフローチャート。

【図 31】

適用 Z 値算出処理（処理 1005）の一例を示したフローチャート。

【図 32】

視差値算出処理（処理 1006）の一例を示したフローチャート。

【図 3 3】

本発明の他の実施例にかかる立体視画像表示装置の一例を示したブロック図。

【図 3 4】

立体視画像表示装置で、動画を受信して表示する場合の処理の一例を示したフローチャート。

【図 3 5】

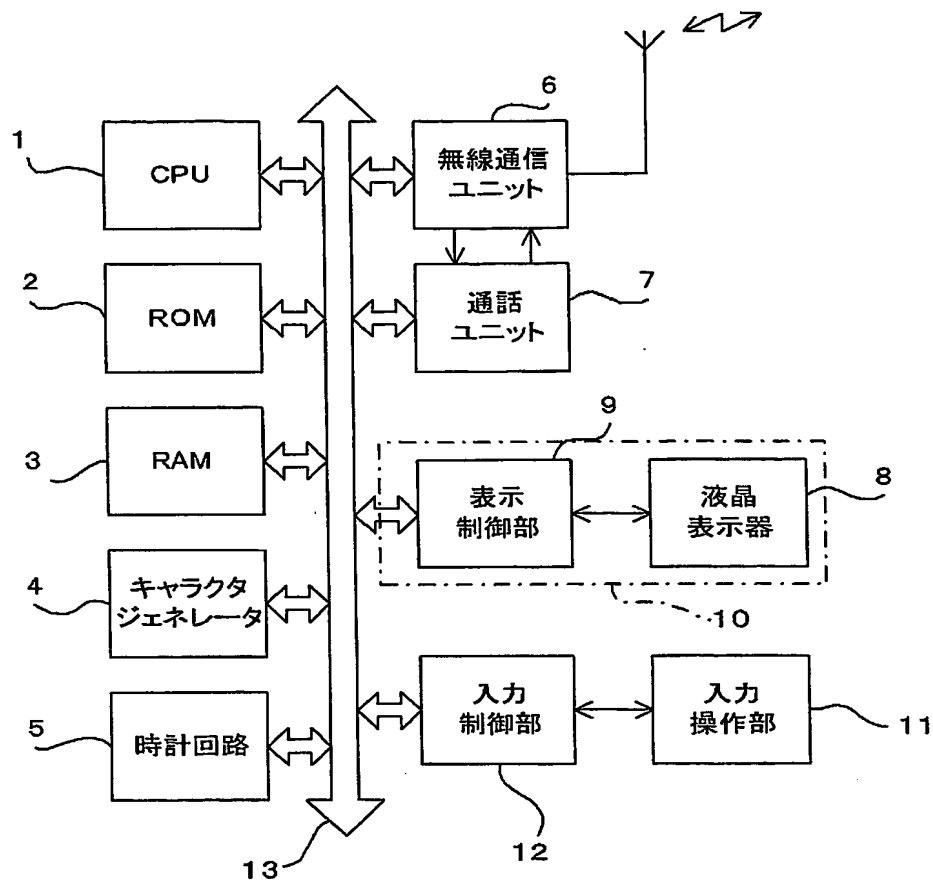
本発明のさらに他の実施例にかかる運転シミュレータ装置の一例を示したブロック図。

【符号の説明】

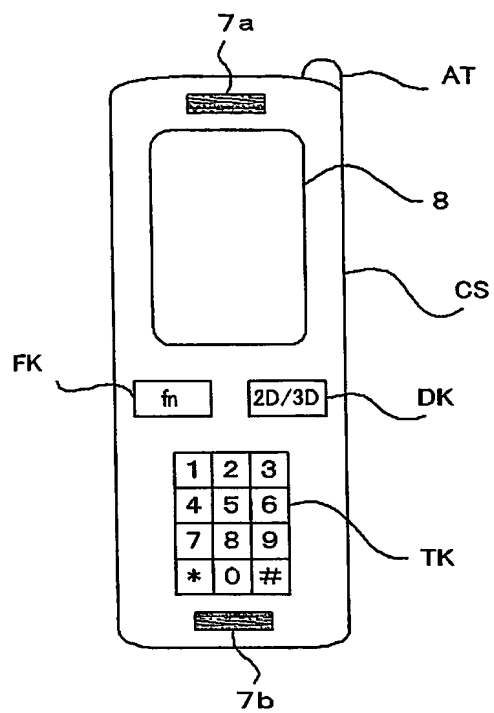
- 1, 21 CPU (中央処理装置)
- 2, 22 ROM (リード・オンリ・メモリ)
- 3, 23 RAM (ランダム・アクセス・メモリ)
- 8 液晶表示器
- 9, 34 表示制御部
- 10 立体視表示装置
- 35 左目画像データバッファ
- 36 右目画像データバッファ
- 37 切換器
- 38 画面表示装置
- DK 切換スイッチ

【書類名】 図面

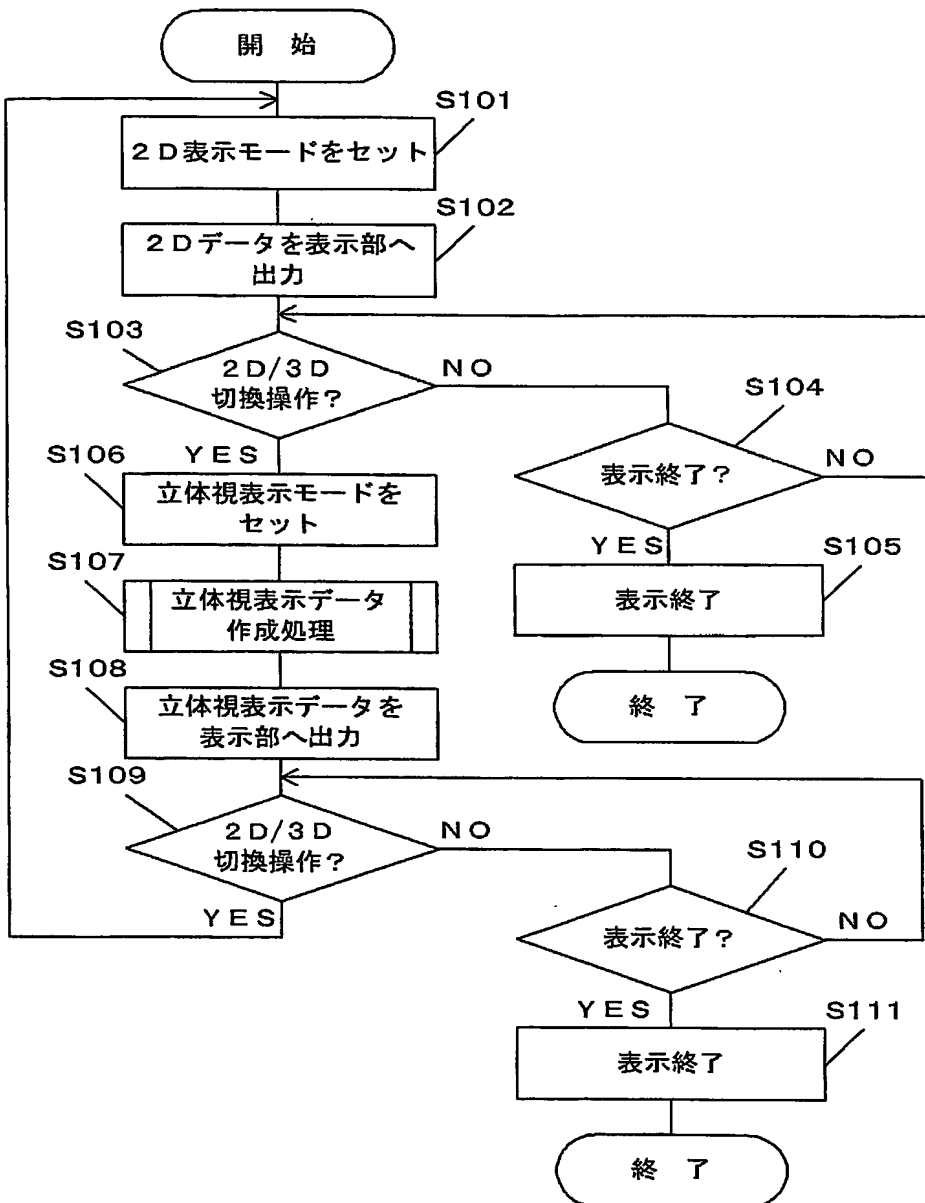
【図 1】



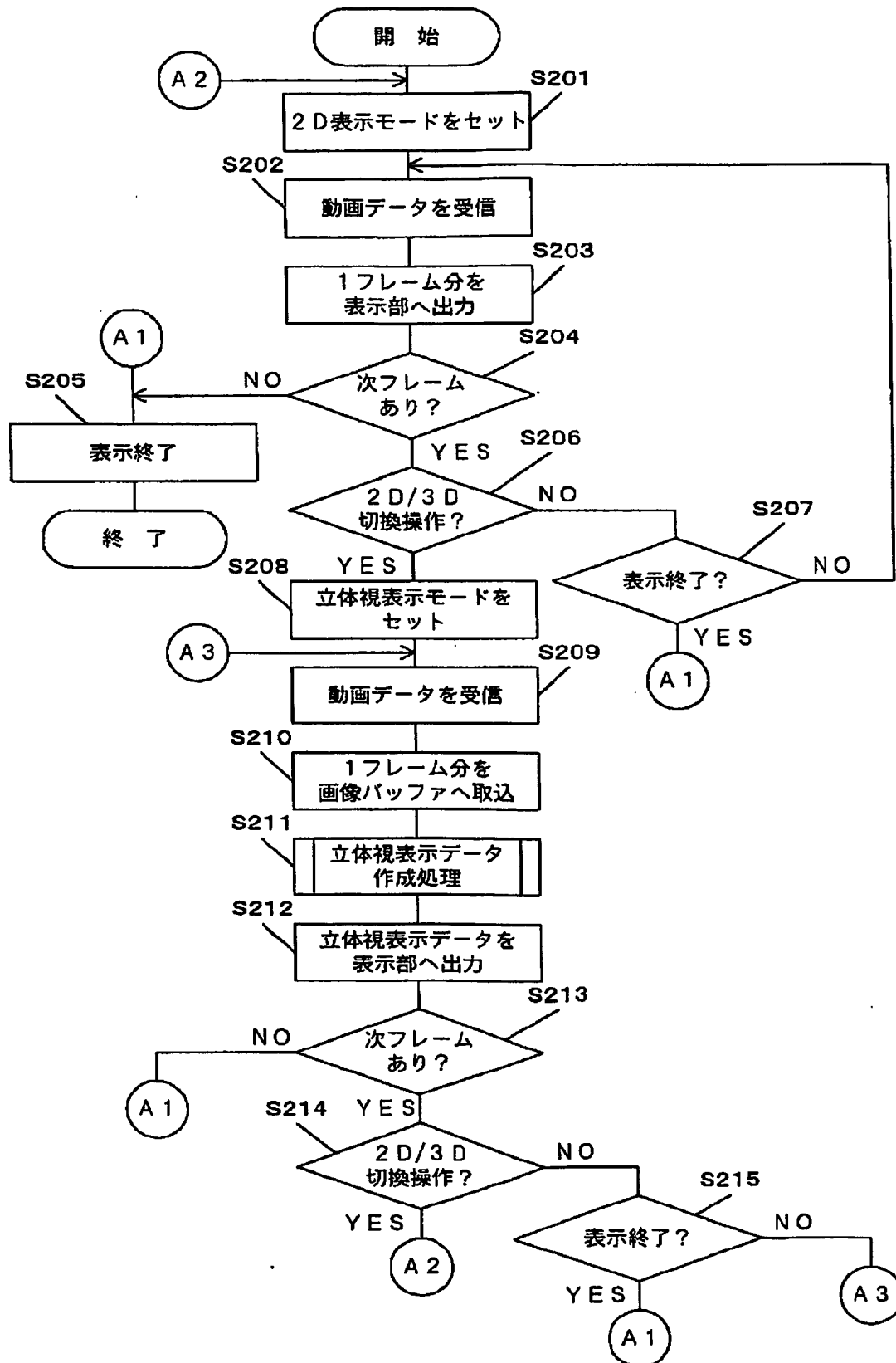
【図 2】



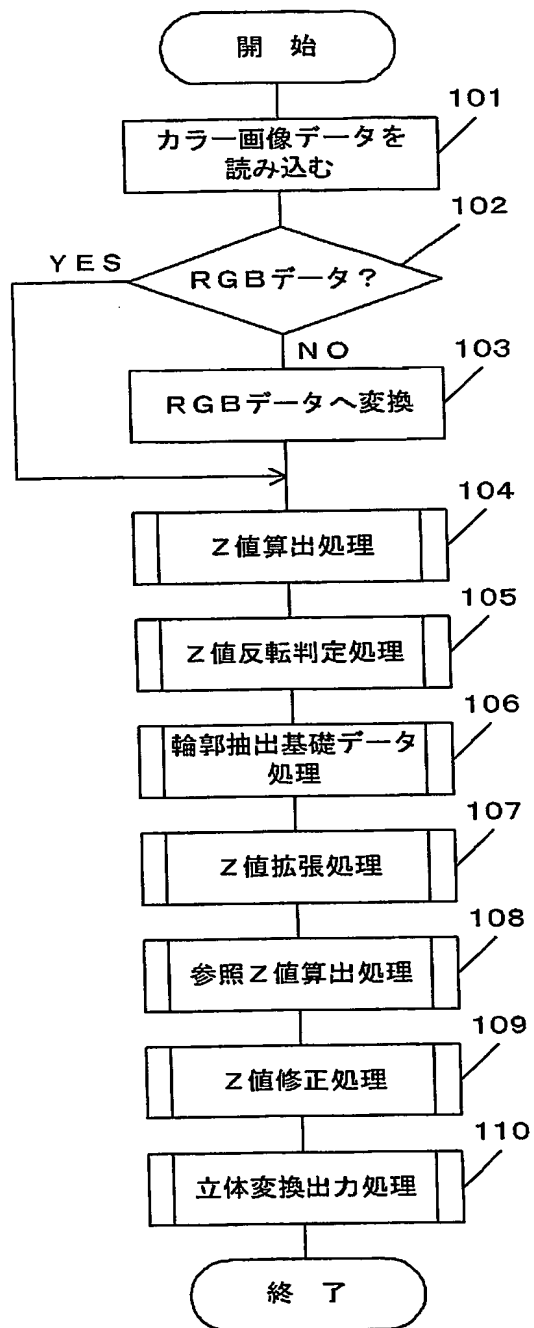
【図 3】



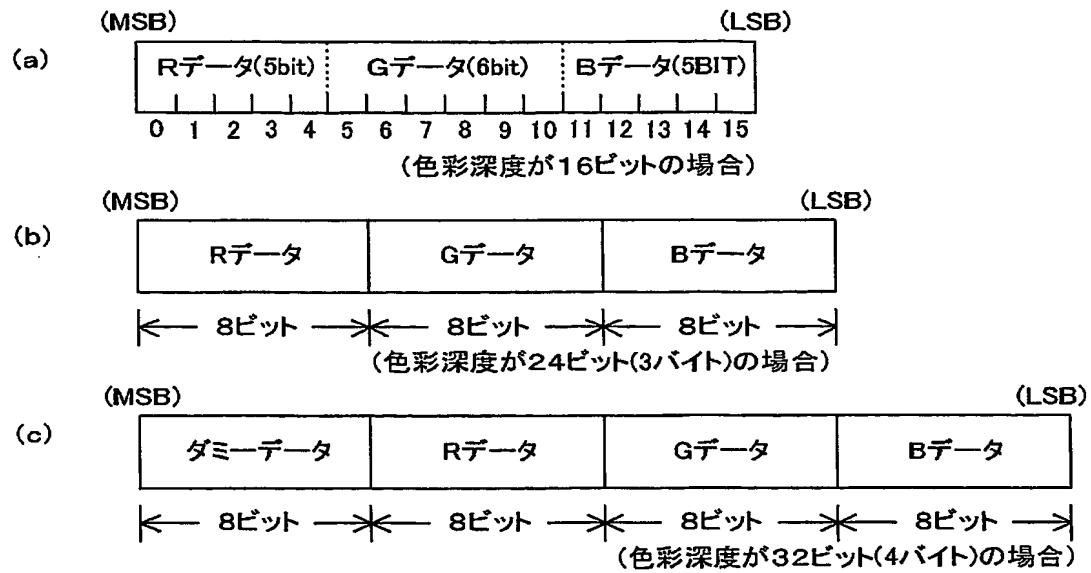
【図 4】



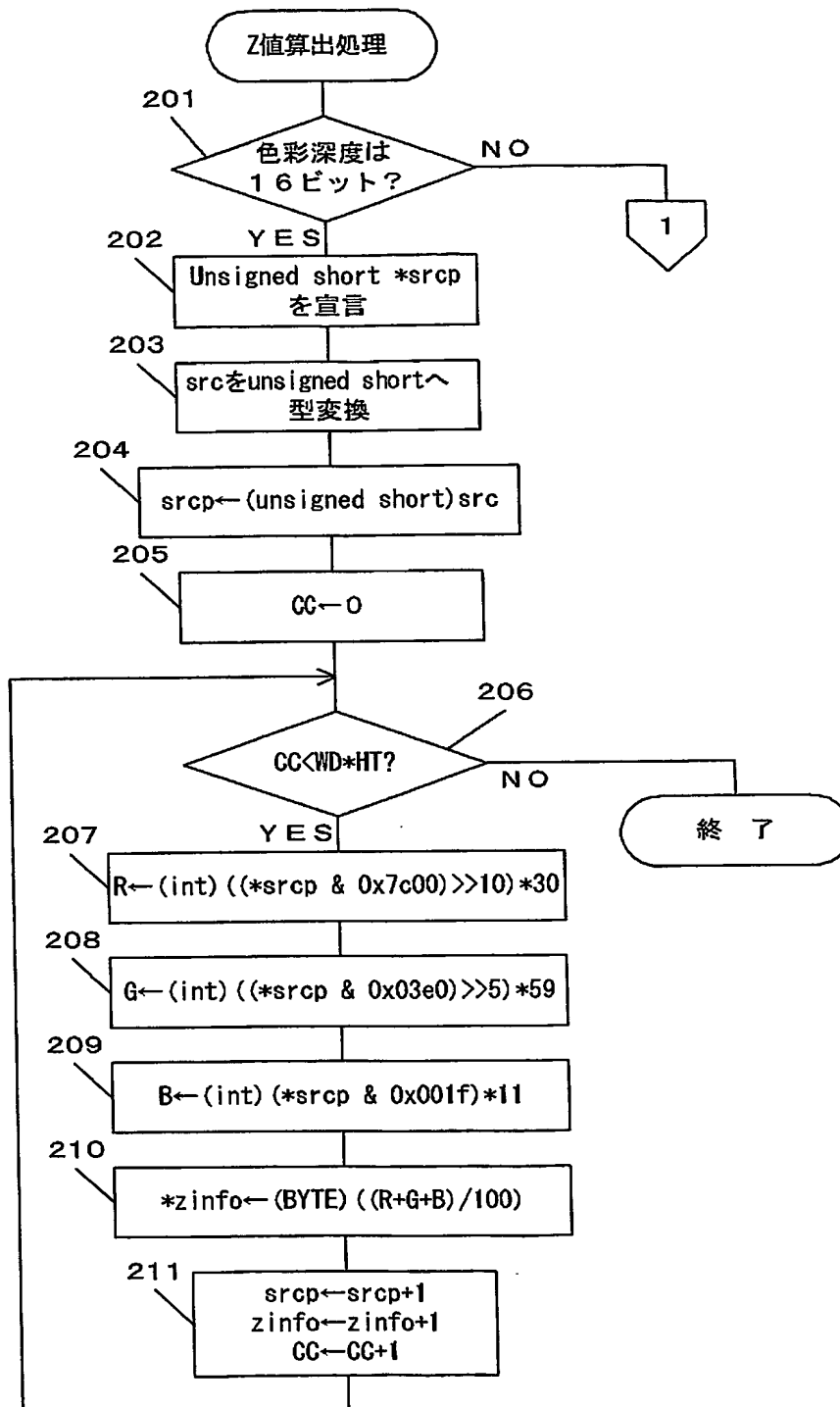
【図 5】



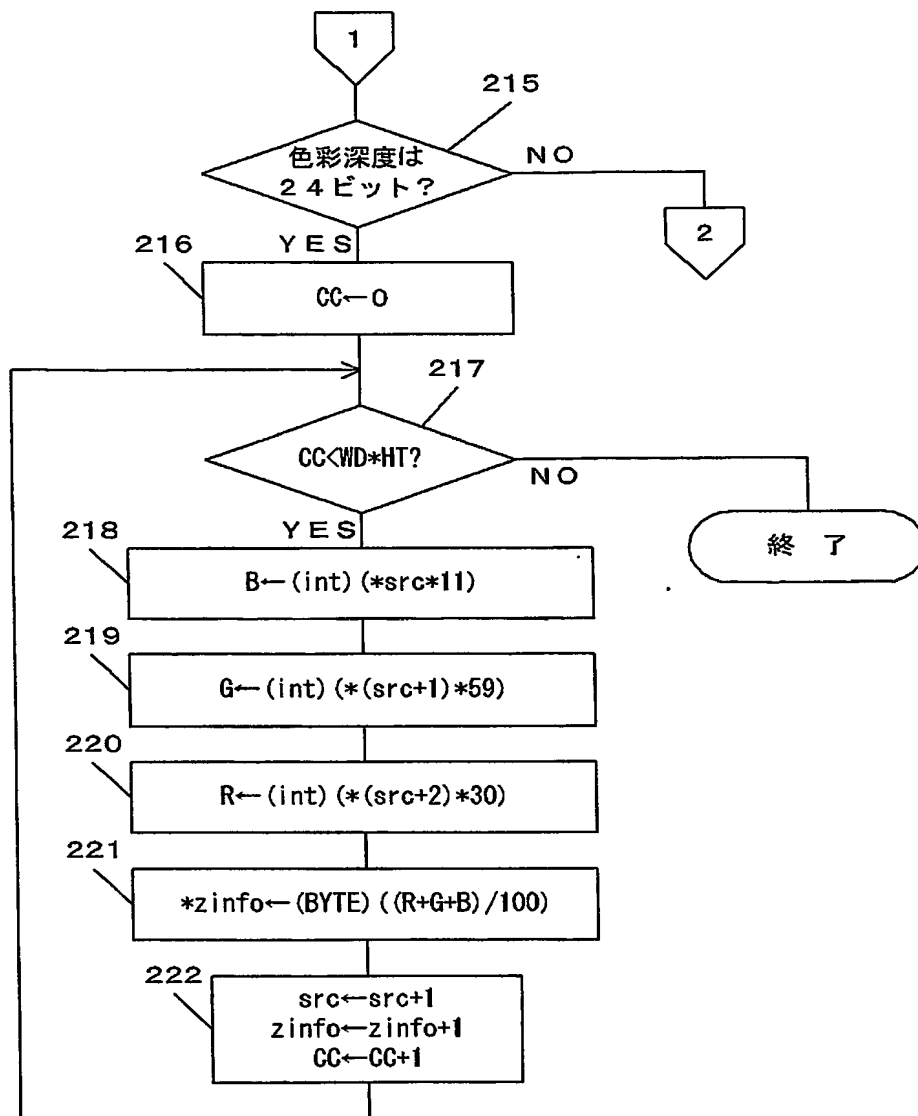
【図 6】



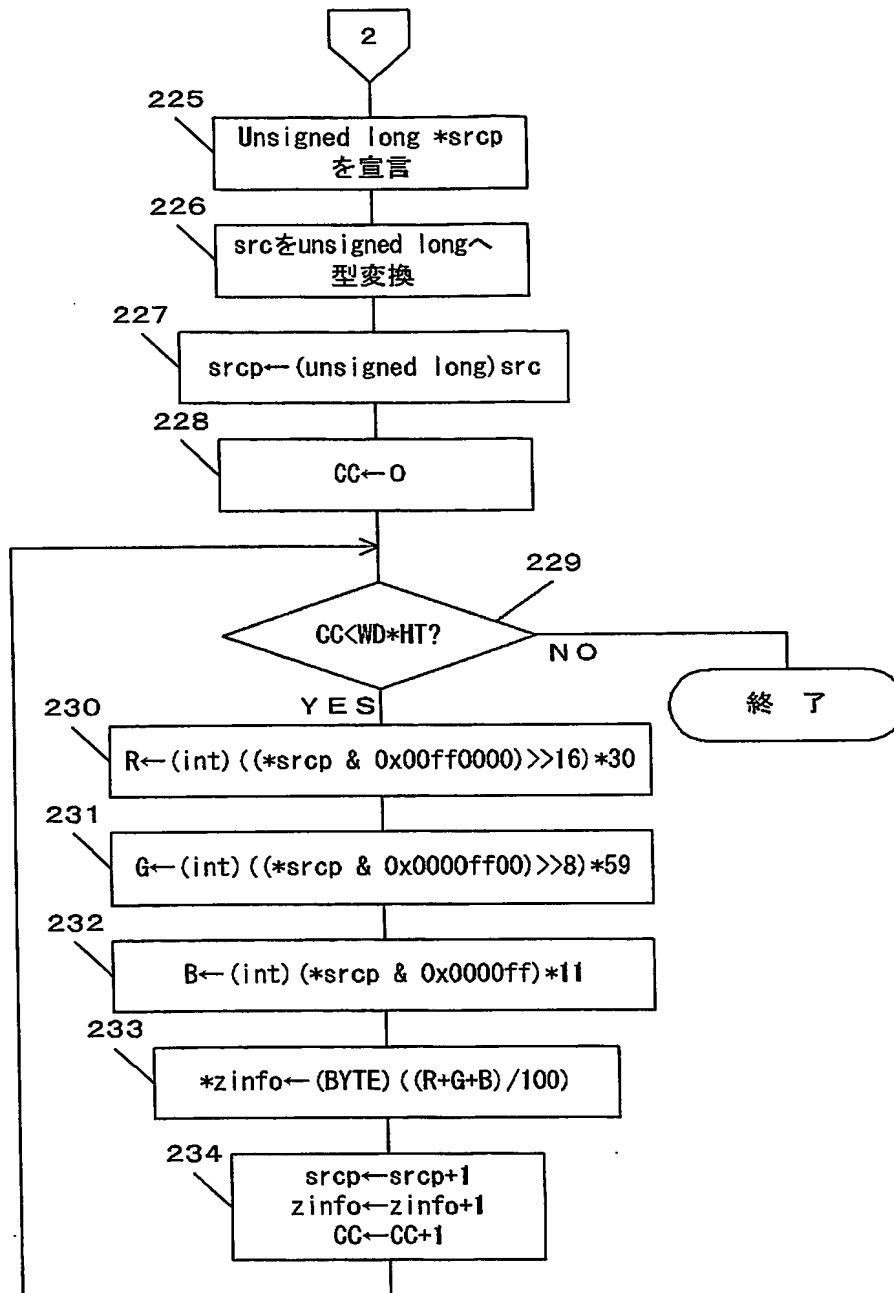
【図7】



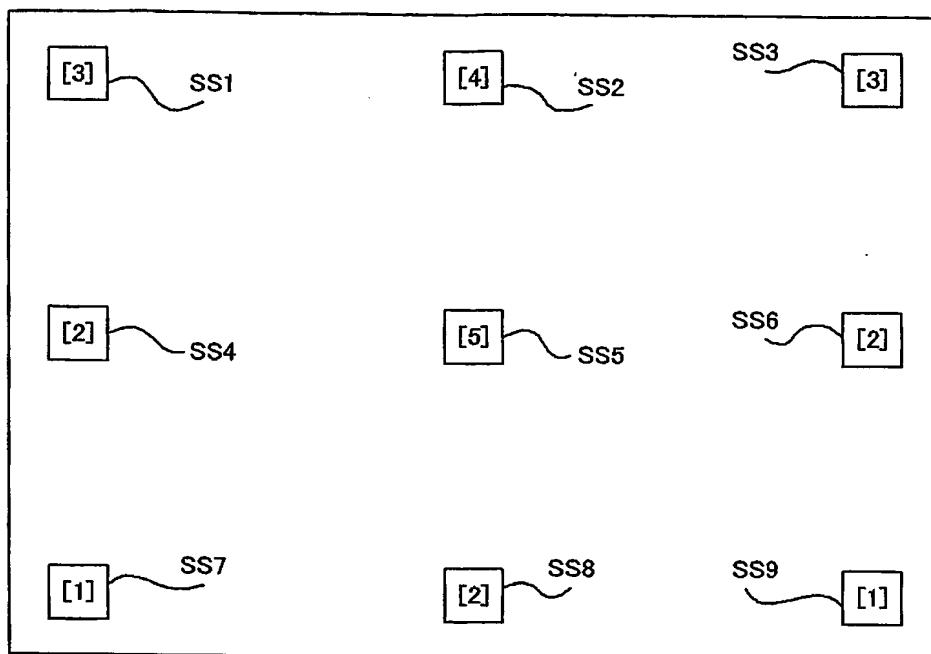
【図 8】



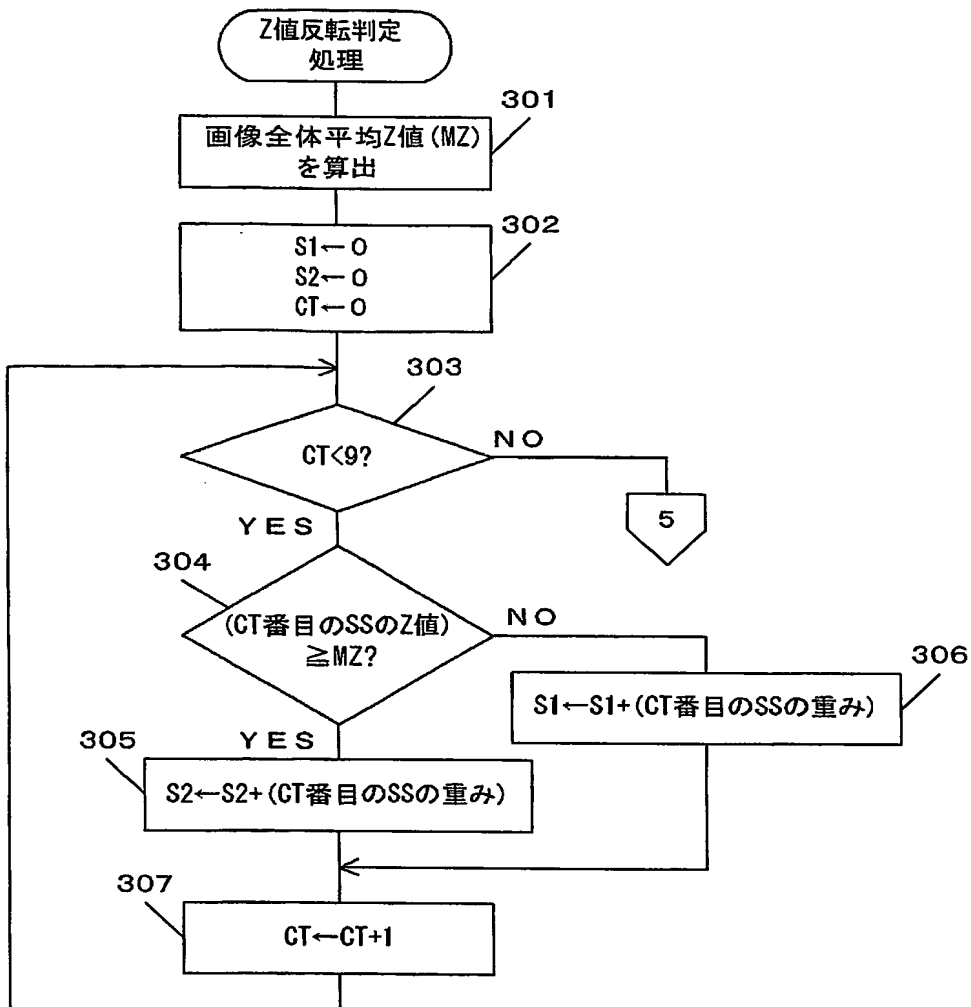
【図 9】



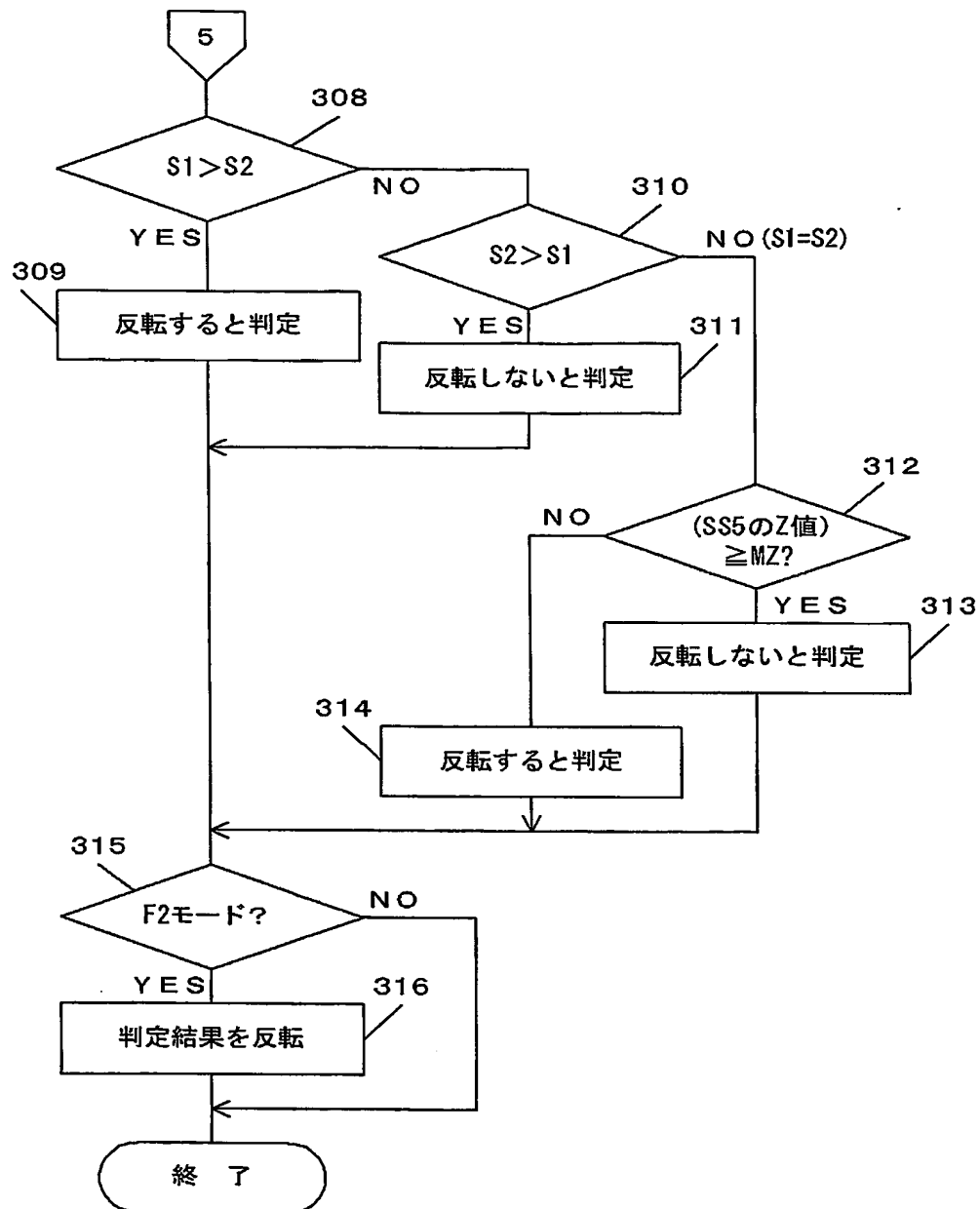
【図 10】



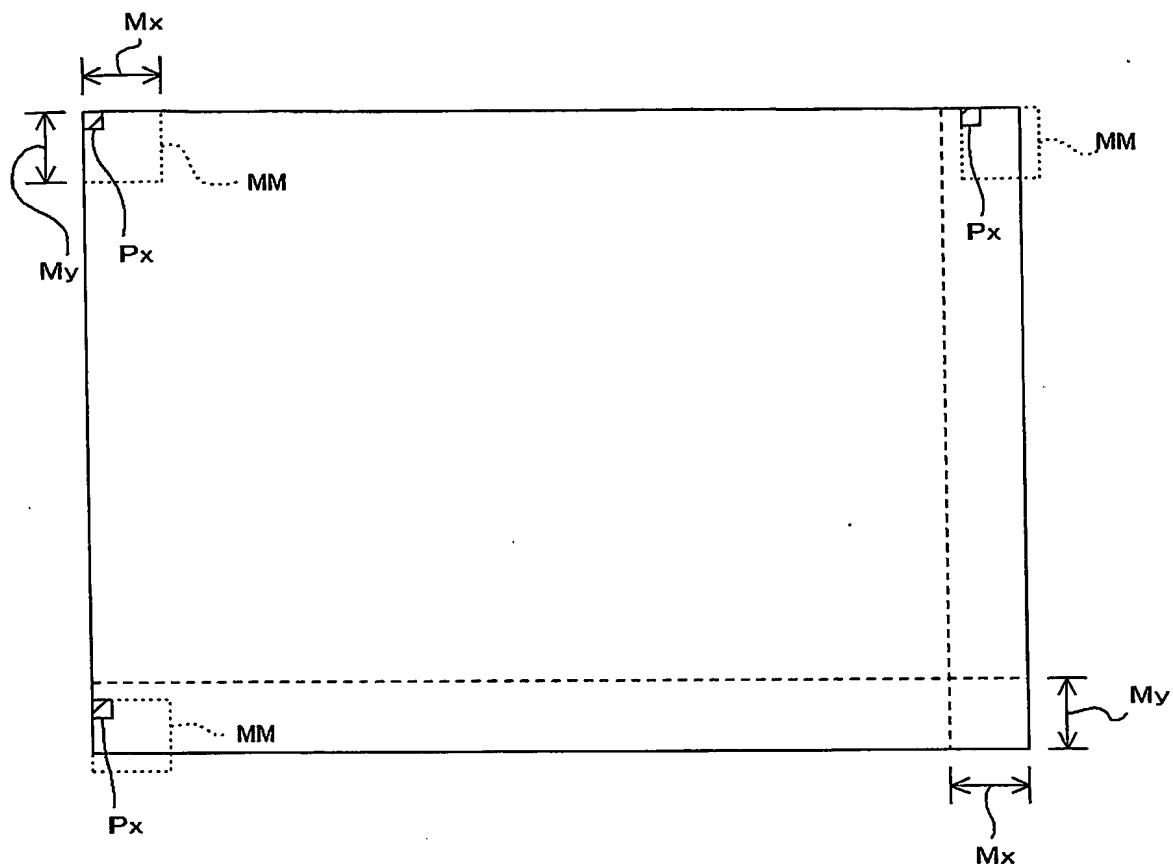
【図 11】



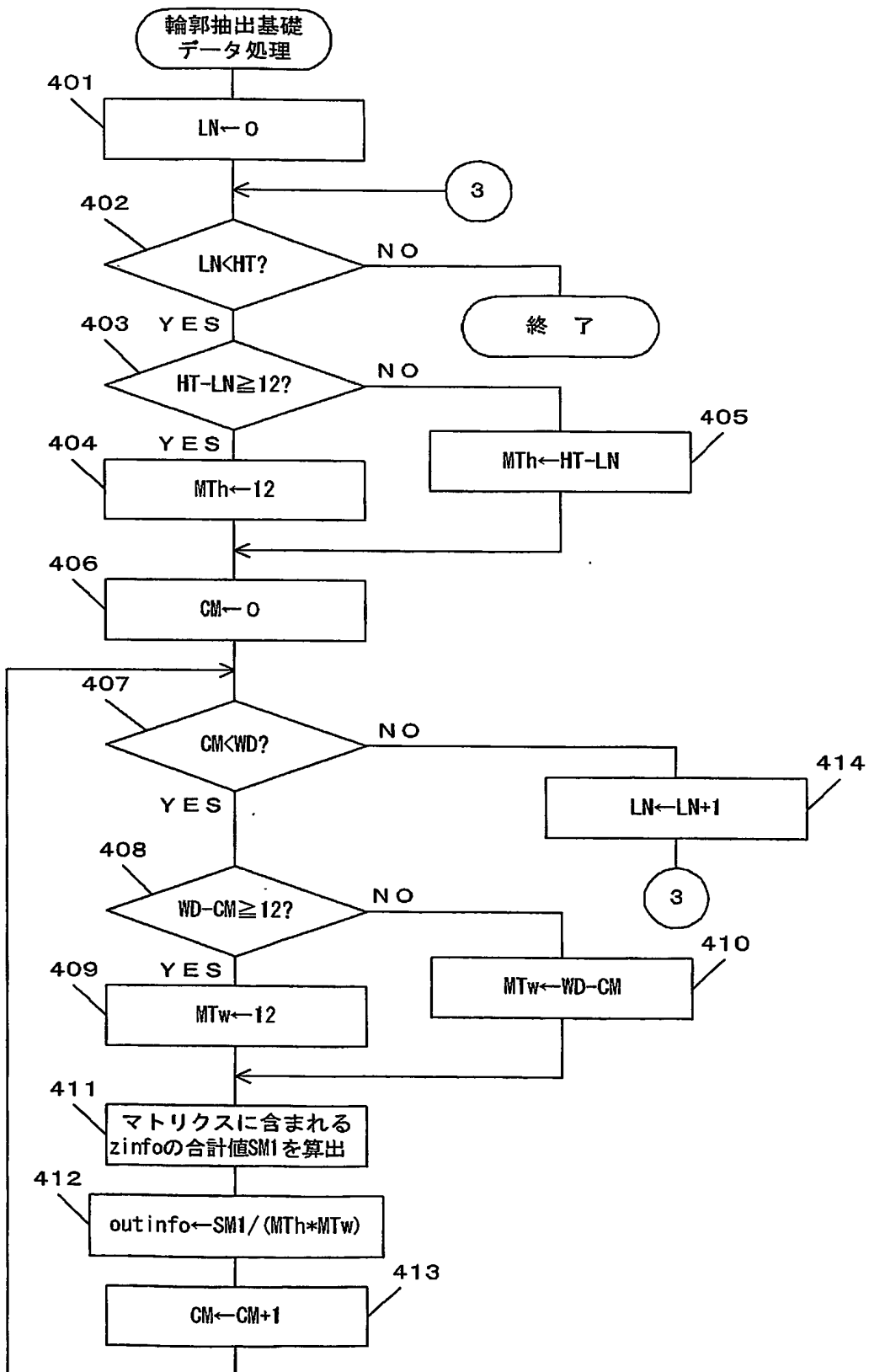
【図 12】



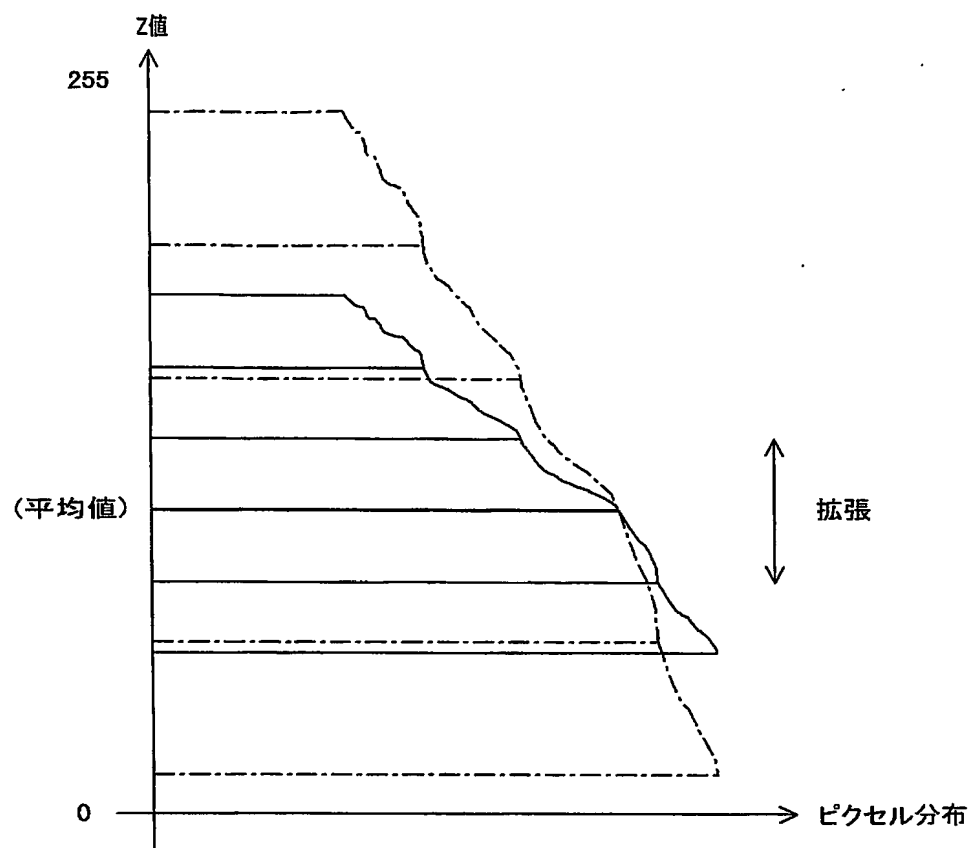
【図 13】



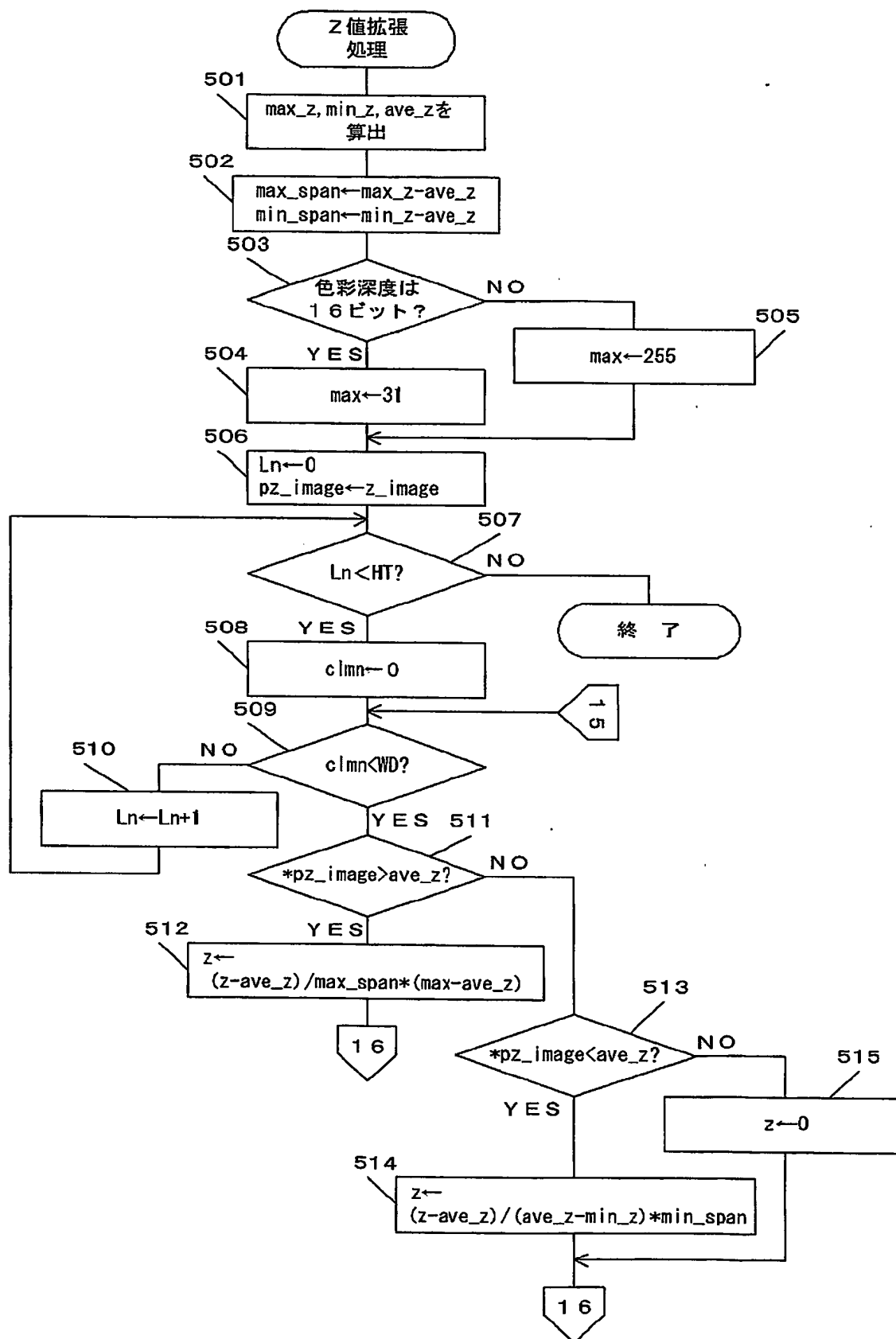
【図 14】



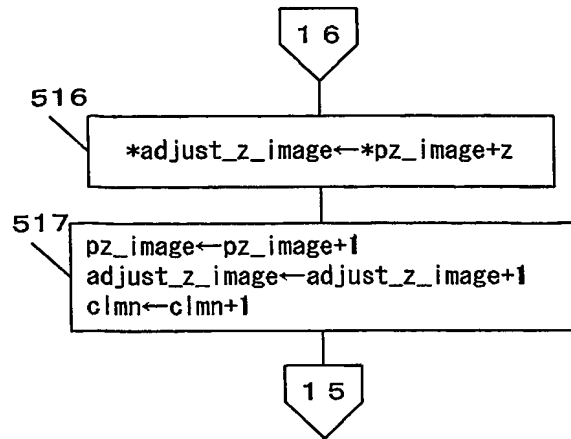
【図 15】



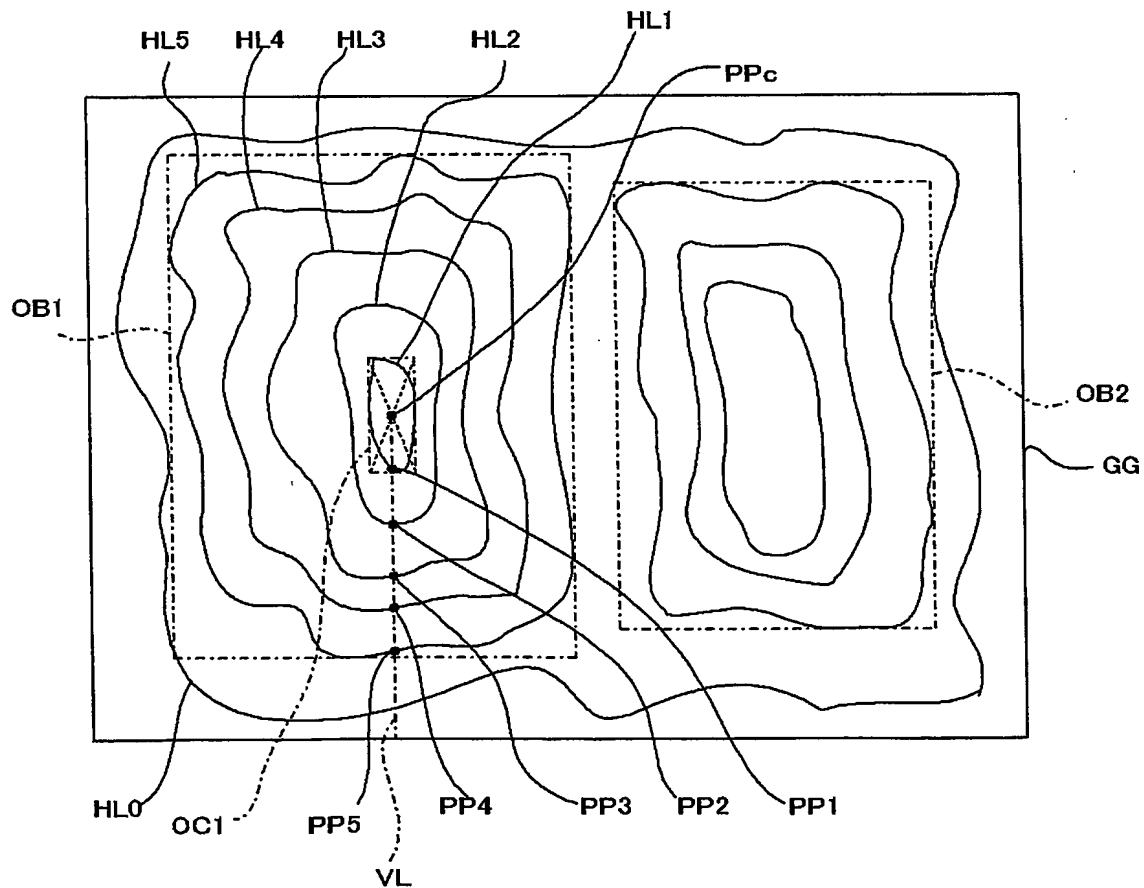
【図 16】



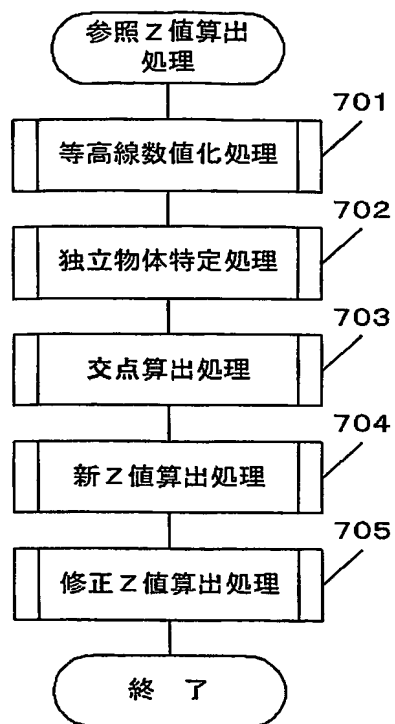
【図 17】



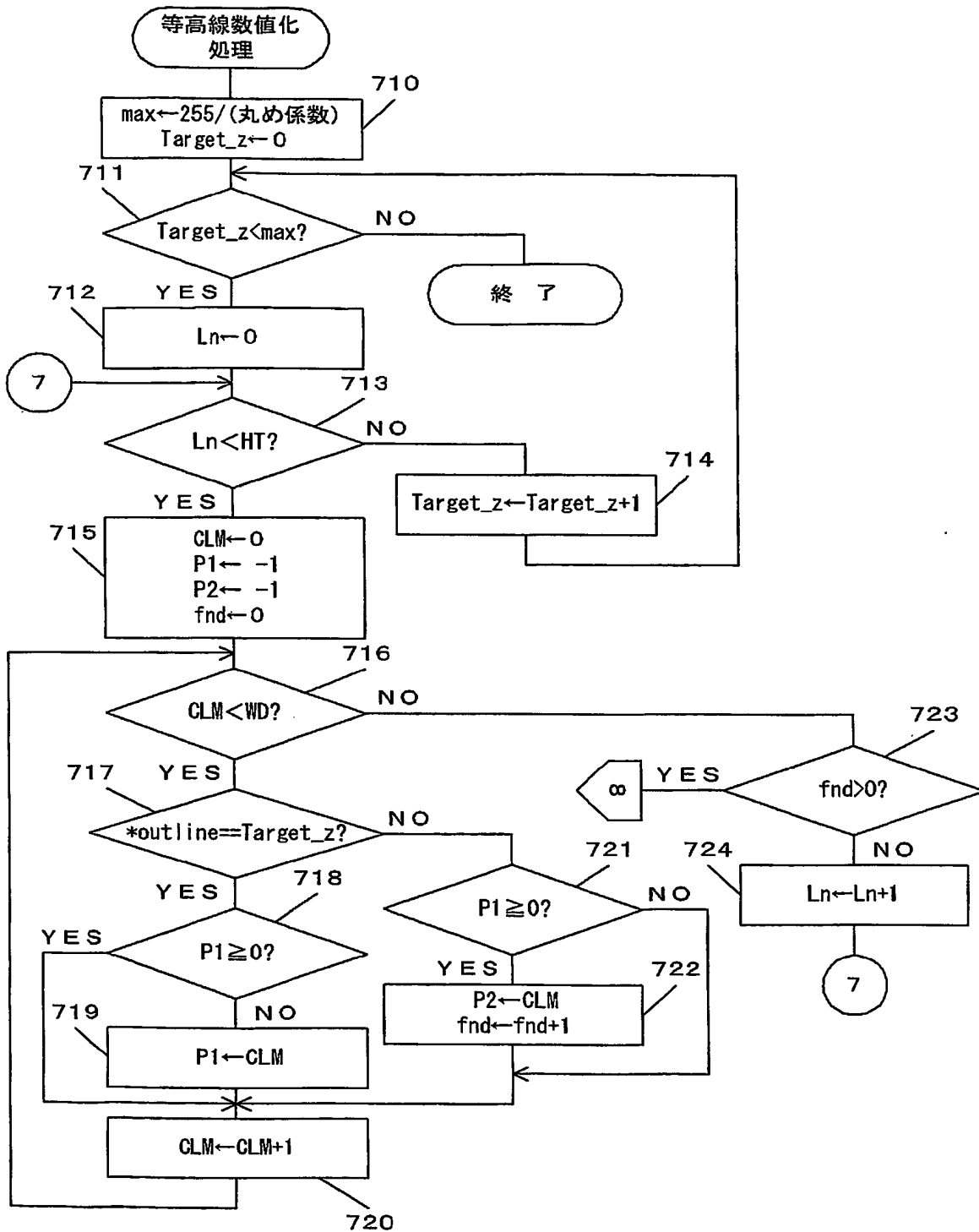
【図 18】



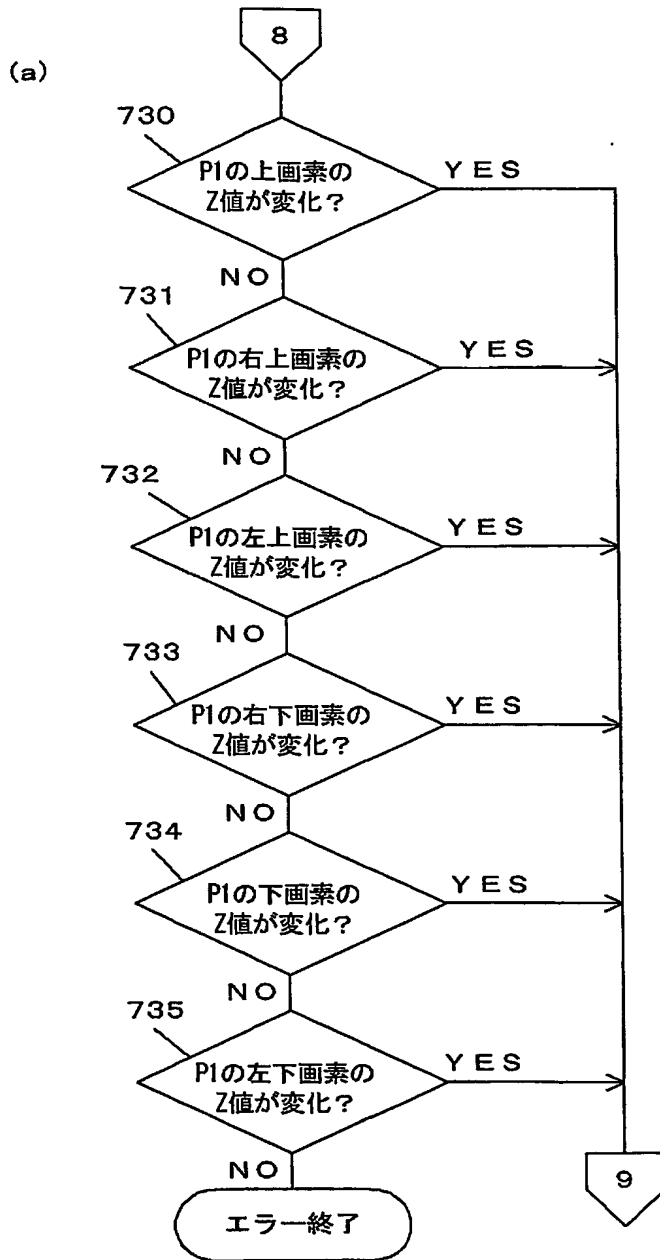
【図 19】



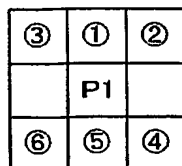
【図 20】



【図 21】

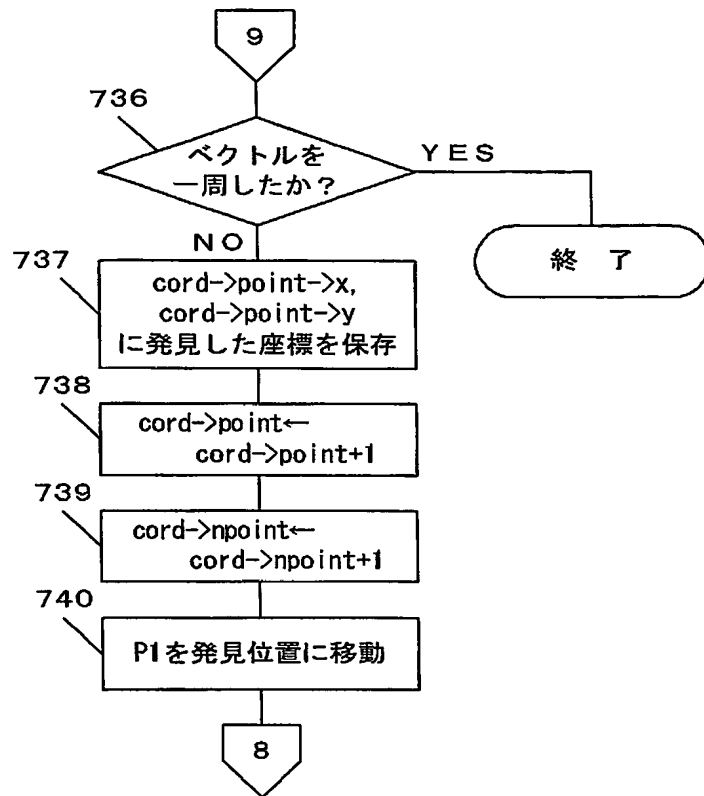


(b)

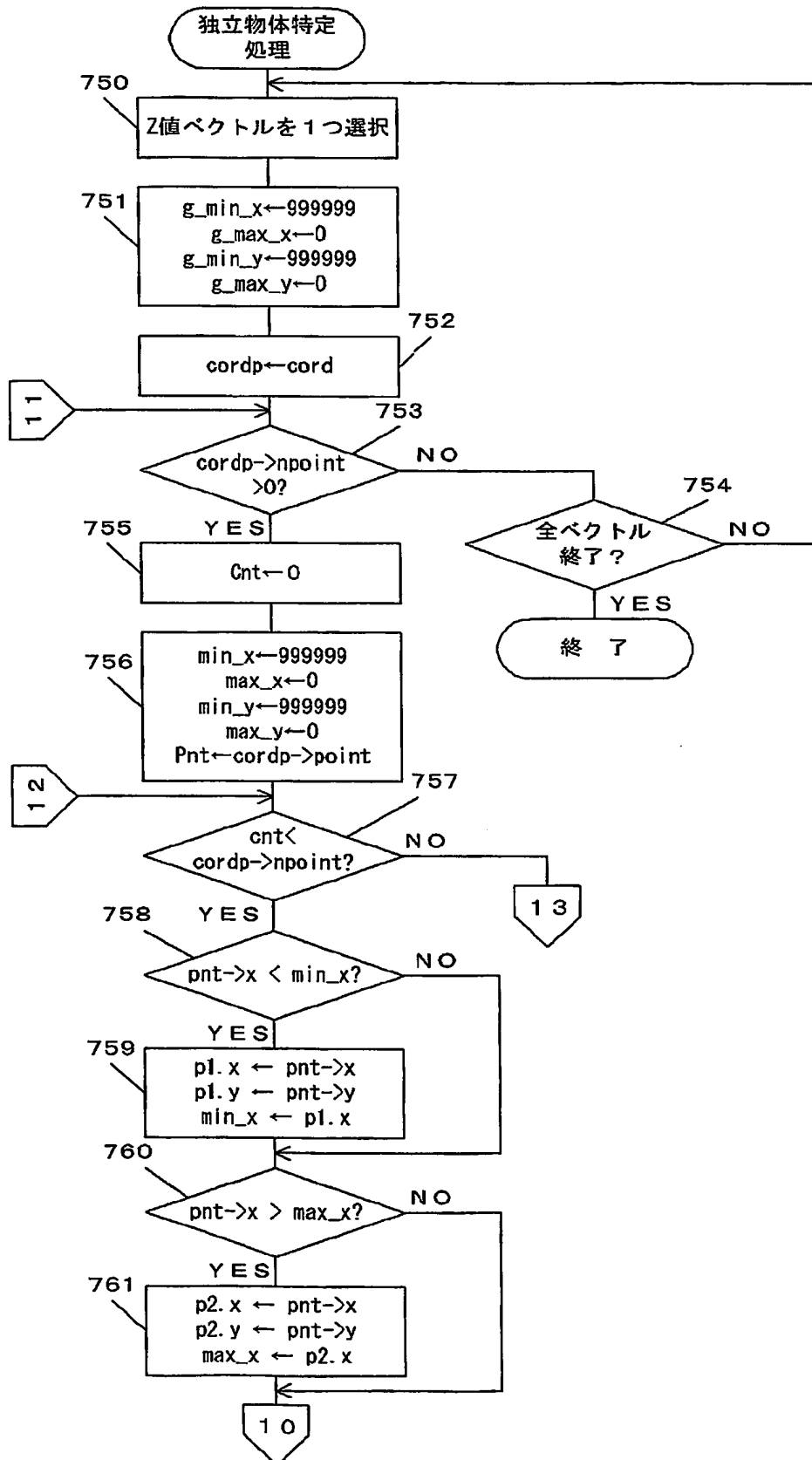


(Z値変化画素探索順序)

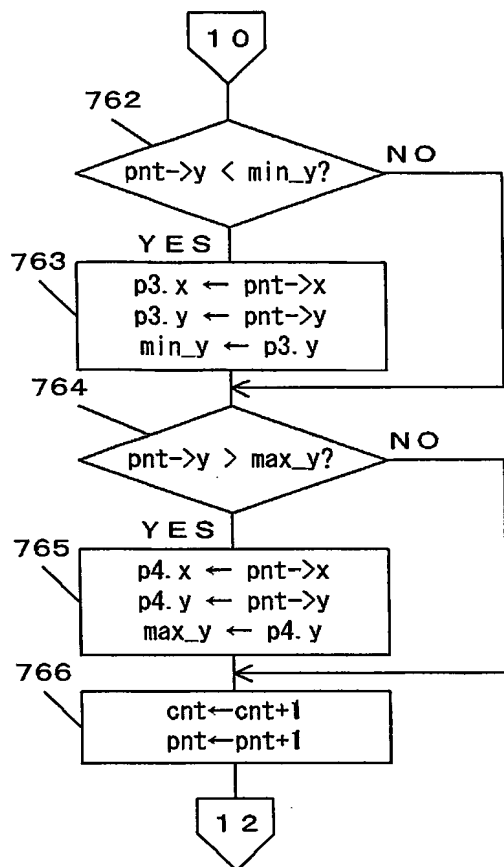
【図 22】



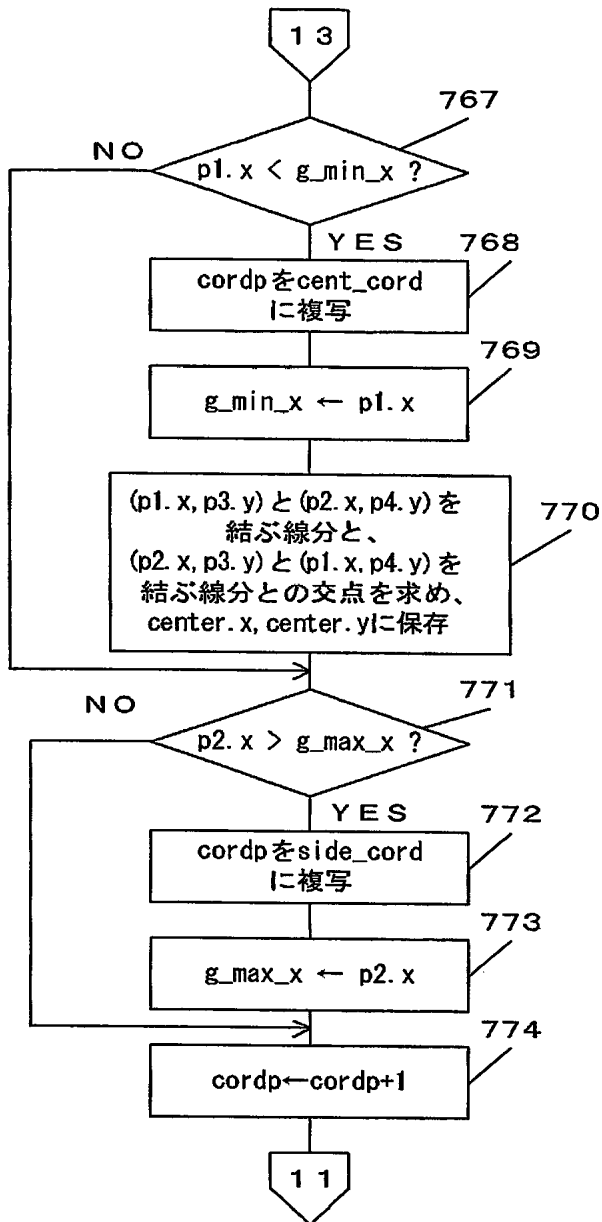
【図 23】



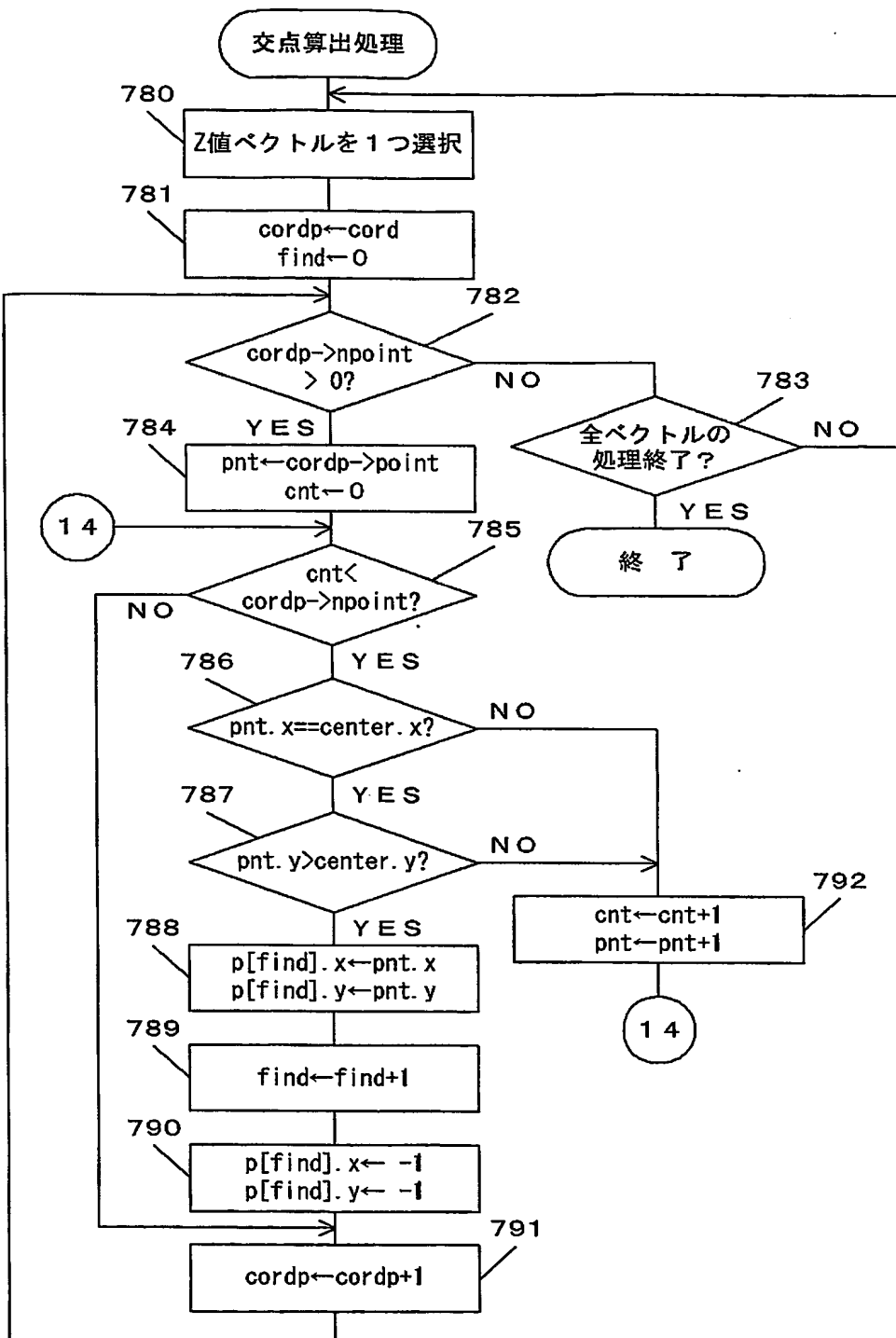
【図 24】



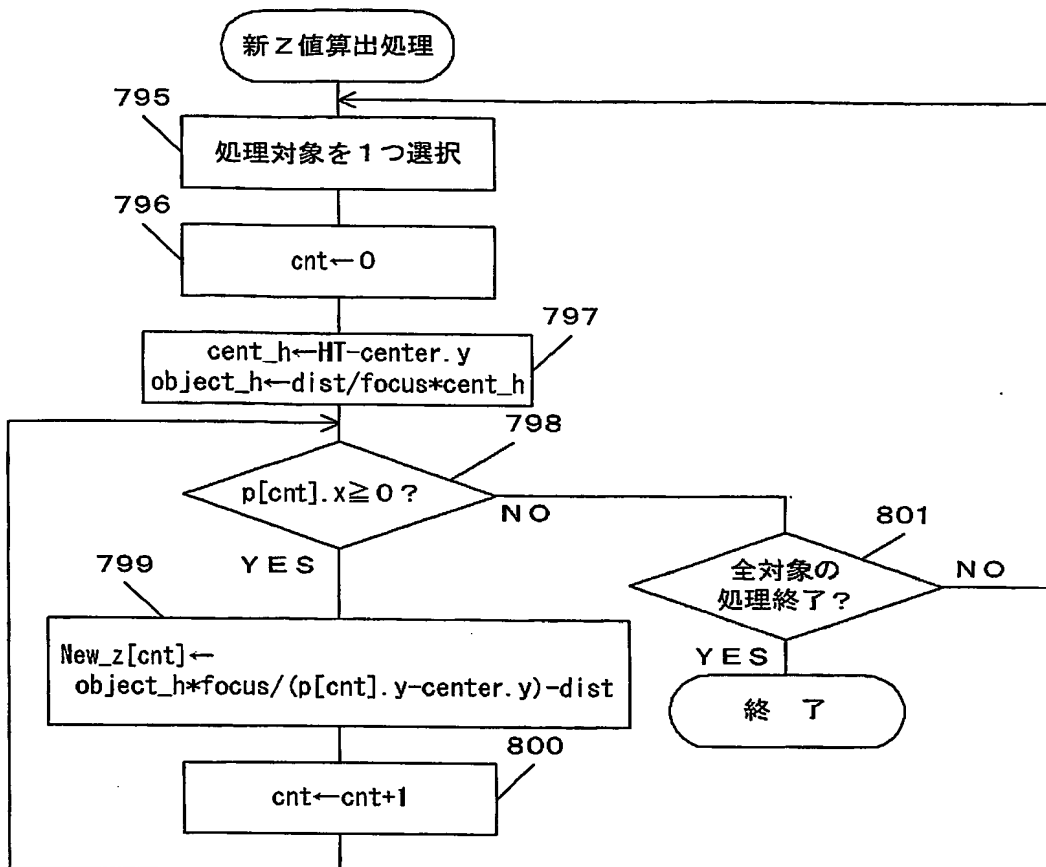
【図 25】



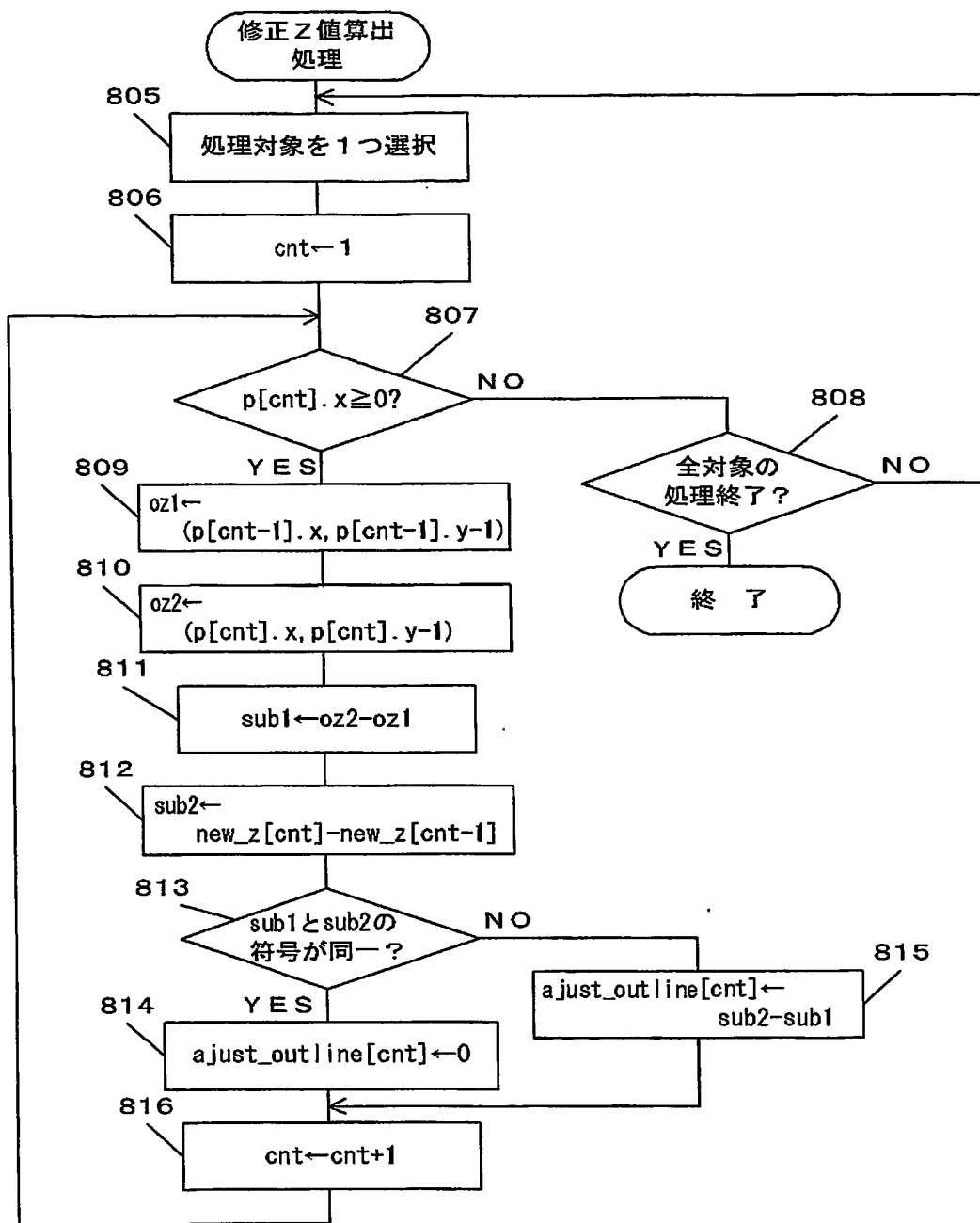
【図 26】



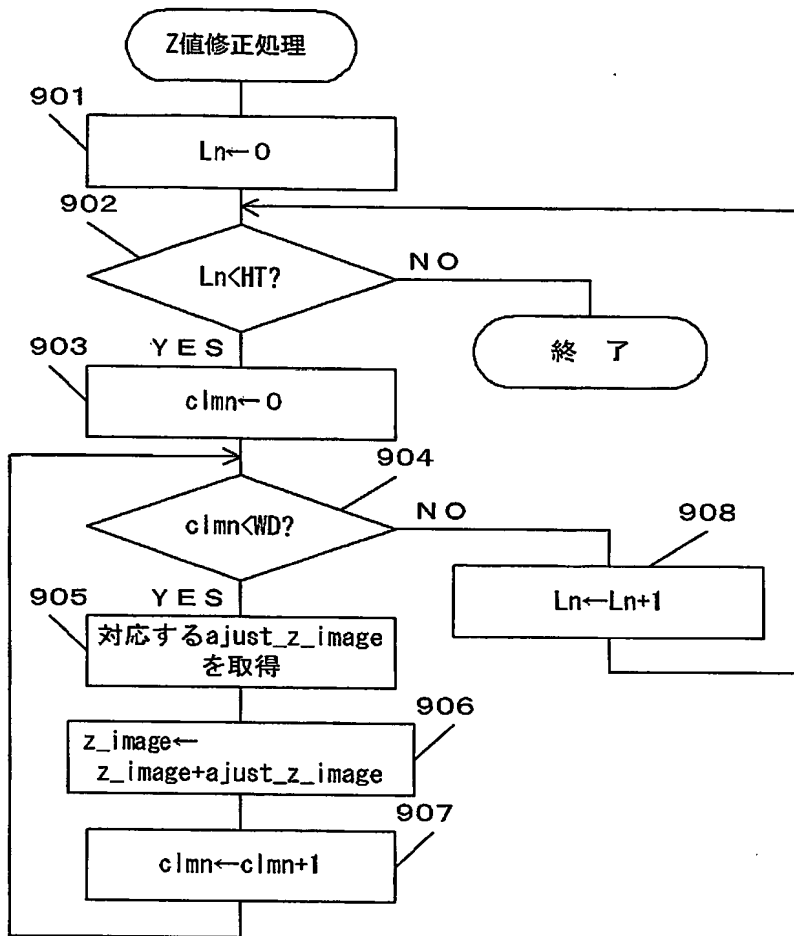
【図 27】



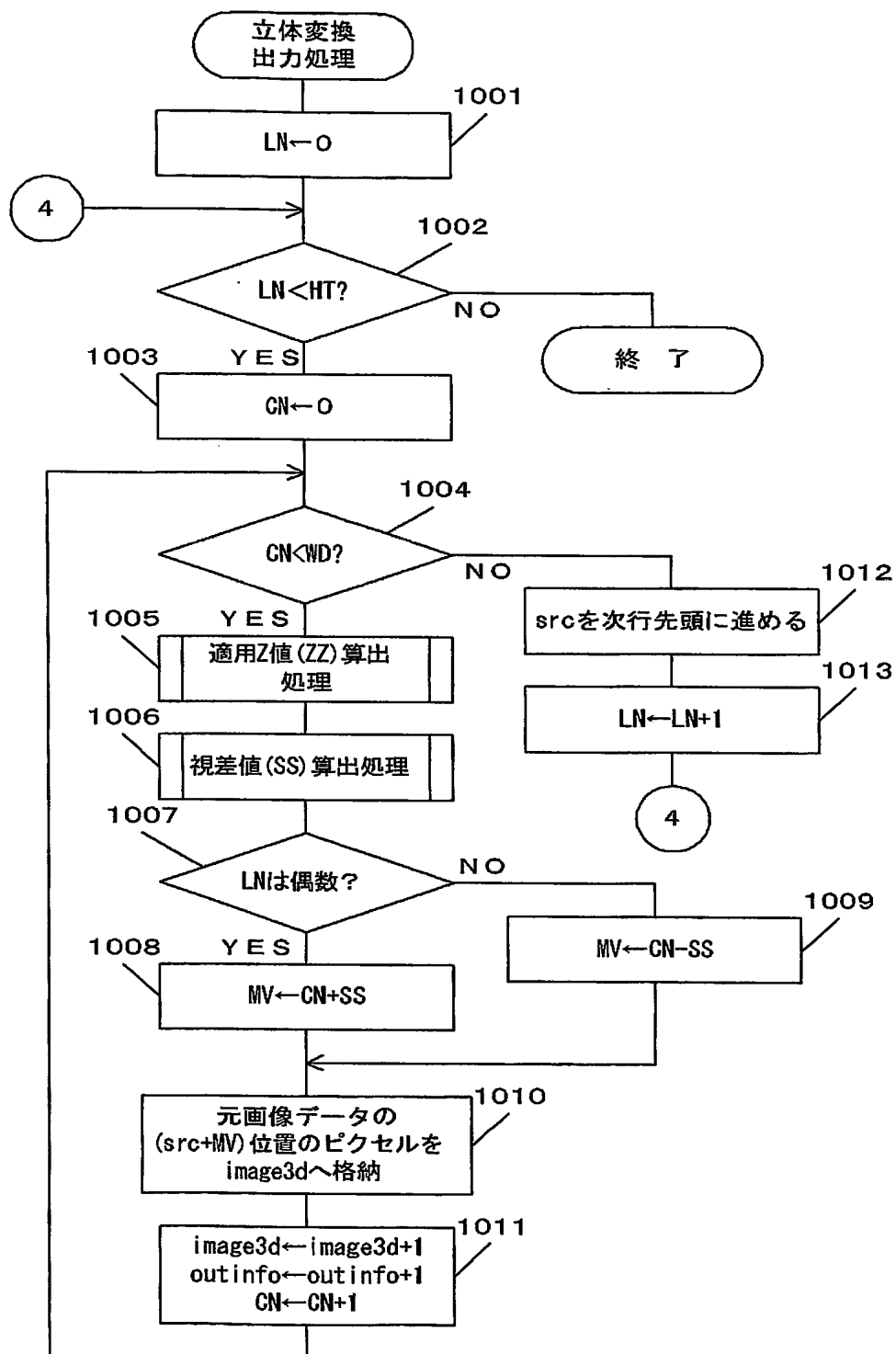
【図 28】



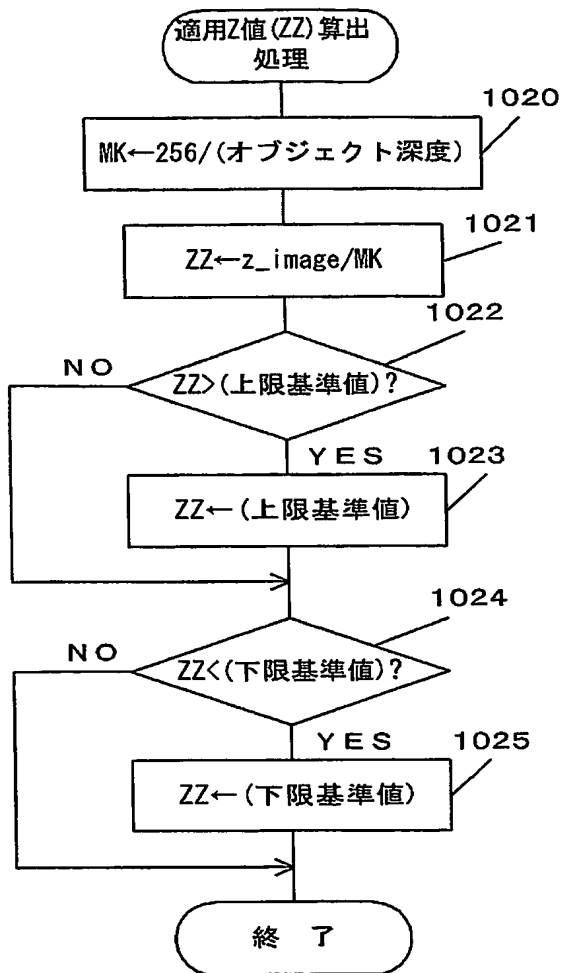
【図 29】



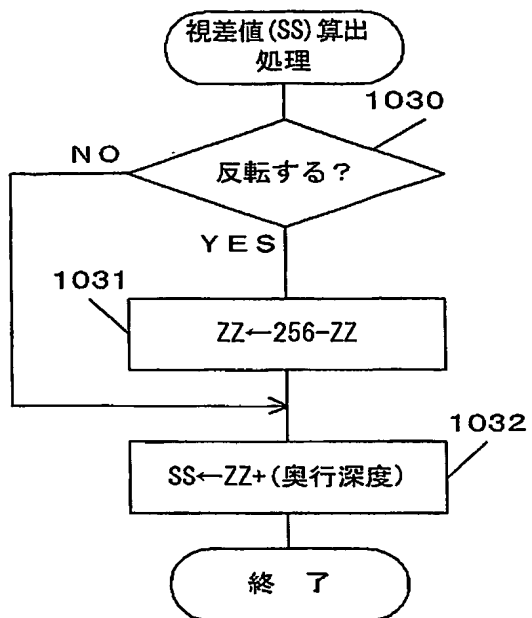
【図 30】



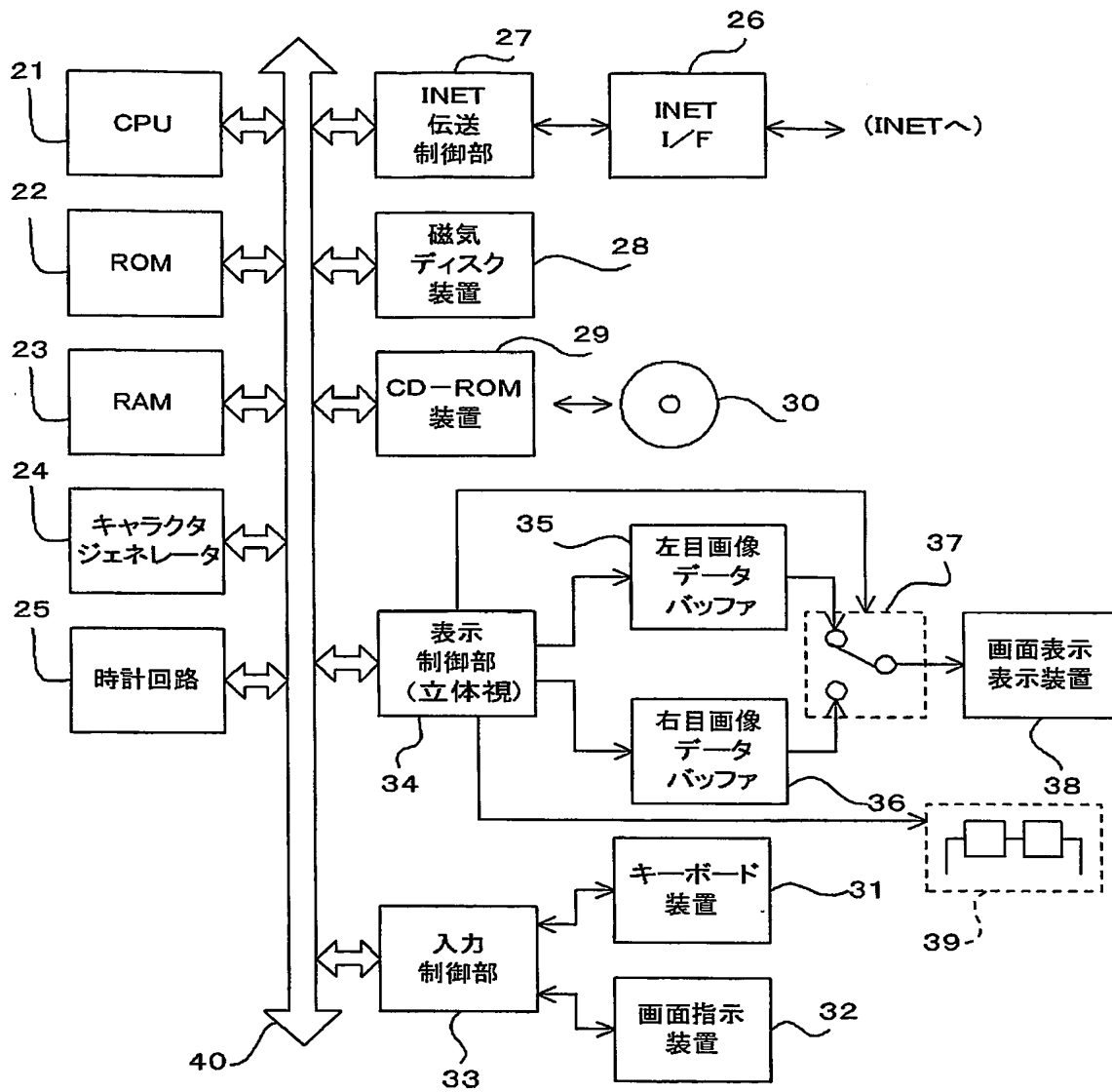
【図 31】



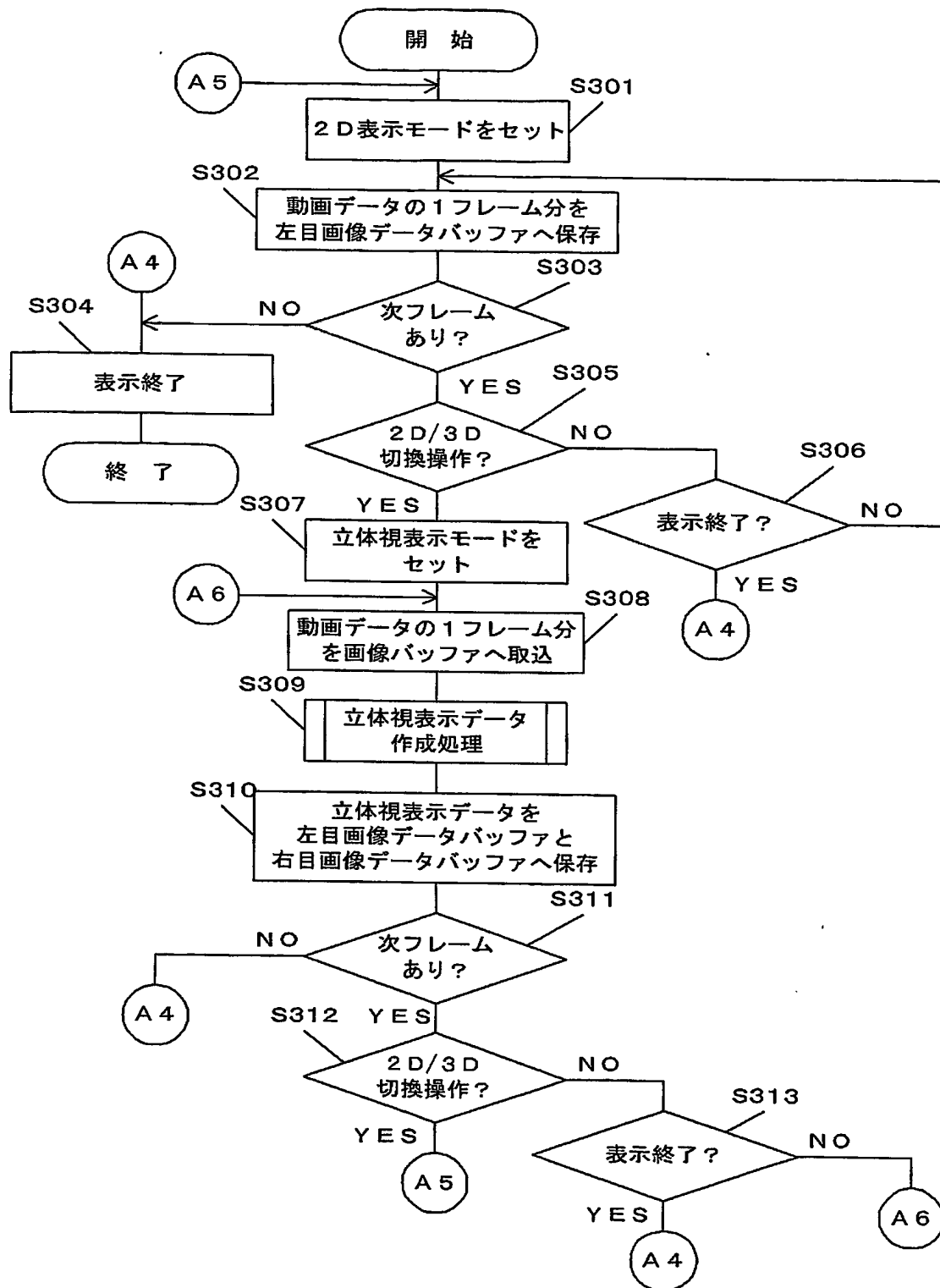
【図 3 2】



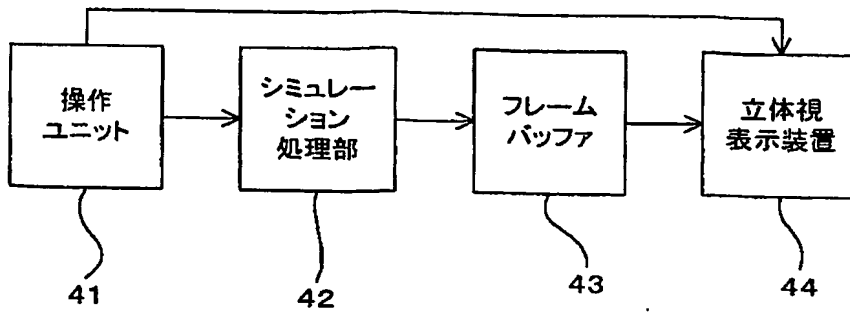
【図 33】



【図 34】



【図 35】



【書類名】 要約書

【要約】

【課題】 立体視画像と非立体視画像を適宜に表示することのできる画像表示装置を提供することを目的としている。

【解決手段】 ユーザが切換スイッチDKを切換操作するたびに、液晶表示器8に表示される画像（静止画または動画）が、非立体視モードの静止画から立体視表示モードの静止画像に、また、立体視表示モードの静止画から非立体視表示モードの静止画像に切り替わるので、非常に便利である。

【選択図】 図2

特願2002-238710

出願人履歴情報

識別番号

[596030737]

1. 変更年月日

1996年 1月31日

[変更理由]

新規登録

住 所

千葉県柏市松葉町4-1-11-201

氏 名

江良 一成

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☒ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.